www.cyberok.ru
info@cyberok.ru
+7 (495) 137-7337

Как бы я взломал Рунет?

В сегодняшней статье мы поделимся результатами более чем годового проекта СайберОК, посвященного анализу защищенности периметра Рунета.

Исследование содержит в себе подробную статистику по некоторым кейсам, аналитические выводы, а также крупные инциденты и опасные уязвимости, выявленные в процессе работы. Особое внимание уделяется не только техническим, но и организационным аспектам реакции на выявленные уязвимости. Помимо этого, в статье приведены технические нюансы, с которыми мы сталкиваемся при реализации подобных масштабных проектов, а также применение таких модных словечек, как LLM, CNN и других, для автоматизации рутины и повышения эффективности работы.

Доклад «Как бы я взломал Рунет» был впервые представлен на международной научно-практической конференции РусКрипто 2024 Игорем Первушиным – руководителем направления создания базы знаний компании CyberOK.

Введение

Для того чтобы что-то взломать, надо сначала выяснить: что же оно такое?

Начнем с масштабов количества ресурсов, которые нам необходимо проверить. Мы в СайберОК собрали следующую статистику при помощи системы контроля и информирования о поверхности атак (СКИПА), которую мы разработали и совершенствуем, а также данных других систем класса Attack Surface Management:

- 47.5M IP-адресов – адресное пространство Рунета.
- 7.2M из них являются «живыми». Подразумевается, что на IP-адресе есть хотя бы один сетевой порт.
- При этом на всех IP-адресах размещен 71.7M сервисов, с которых может быть получена информации, которой, в частности, могут воспользоваться злоумышленники. В данную статистику входят также и сервисы, которые были получены при анализе доменных имен.
- 63.5M – это сервисы, которые работают по протоколу HTTP, но помимо него используются еще более 125 различных протоколов.
- Всего нами было обнаружено более 3000 различных типов программного обеспечения. Если учесть, что на каждом из сервисов размещено более 1 типа программного обеспечения, то цифры с возможными точками атаки на уязвимости возрастают кратно.

Для злоумышленника большая поверхность атаки – это преимущество, потому что чем больше возможностей, тем больше вероятностно, что атака будет успешной.

Далее перед нам встает следующий вопрос: как просеять все это многообразие, чтобы затратить наименьшее количество ресурсов и при этом принести как можно больше «пользы»? Как отсечь все ненужное, чтобы добраться до чего-то драгоценного?

Оптимизация сканирования и автоматизация в поиске уязвимостей: методы CyberOK

Уже на этапе сканирования портов при помощи нашей технологии СКИПА мы столкнулись с фактом, того, что «не все ASM одинаково полезны». Так, согласно данным на 20.03.24, в Shodan и Censys порт 10050 практически отсутствует, при этом нами было найдено порядка 135 тысяч узлов. На данном порту чаще всего размещается Zabbix agent.

www.cyberok.ru
info@cyberok.ru
+7 (495) 137-7337

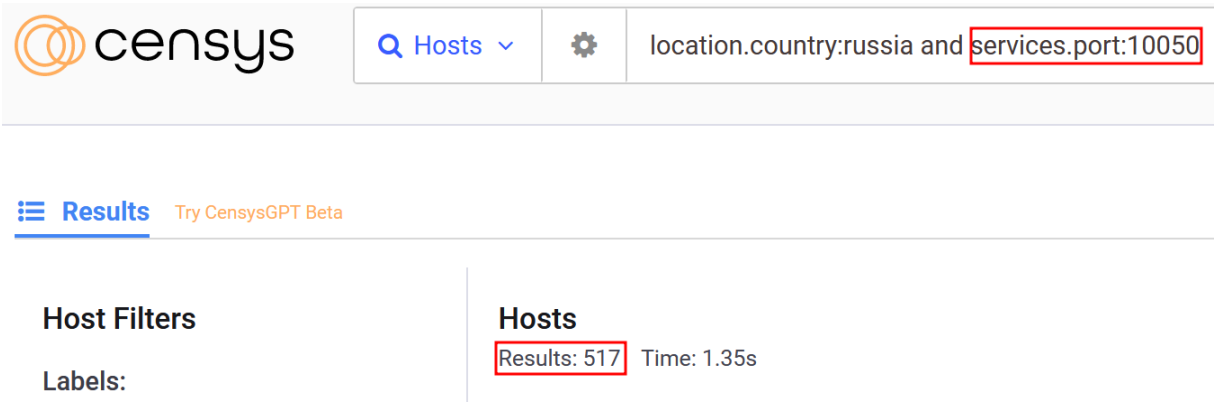

Рис.1 Статистика встречаемости порта 10050 в Рунете в Censys

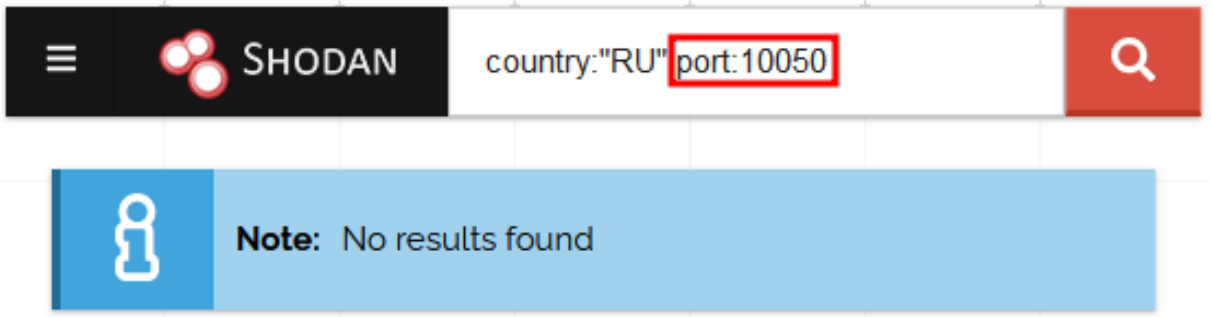

Рис.2 Статистика встречаемости порта 10050 в Рунете в Shodan

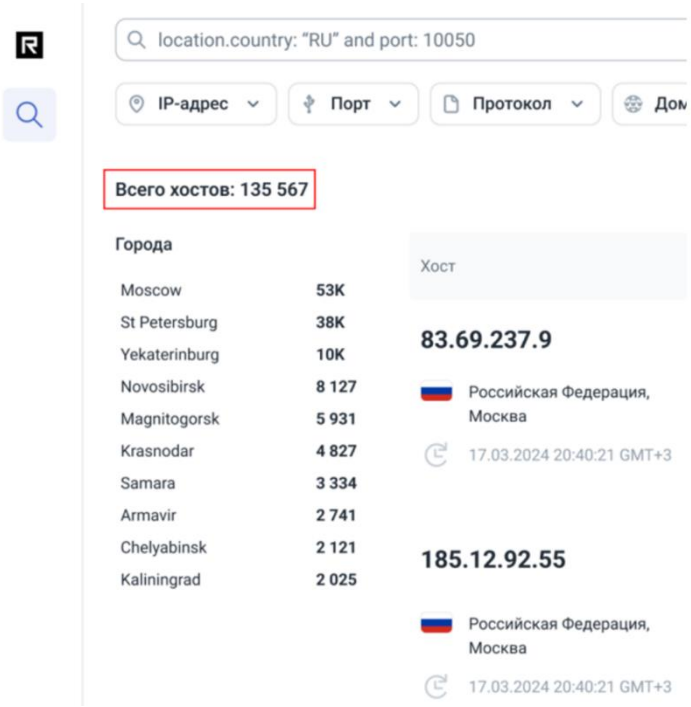

Рис.3 Статистика встречаемости порта 10050 в Рунете в СКИПА

www.cyberok.ru
info@cyberok.ru
+7 (495) 137-7337

В ситуации, когда цифры только растут, необходимо прибегнуть к магии, которая и оптимизирует процесс сканирования, и позволяет сделать за нас большую часть работы. Расскажу о двух из множества методов, которые мы используем.

Метод 1: Оптимизация сканирования TCP-портов

Проверка всех TCP-портов в Рунет довольна дорогая процедура как с точки зрения ресурсов, так и с точки зрения времени. Можно ли как-то оптимизировать процесс, чтобы не стучаться во все двери? На данный момент мы улучшаем свой алгоритм по сканированию IP-адресов на основании предсказания сетевых портов, которые встречаются в рассматриваемом сегменте, что позволяет проверять не все 47.5КК IP-адресов на все 65К портов, а только какую-то их часть. И это гипотеза подтверждается. Посмотрим на картинки:

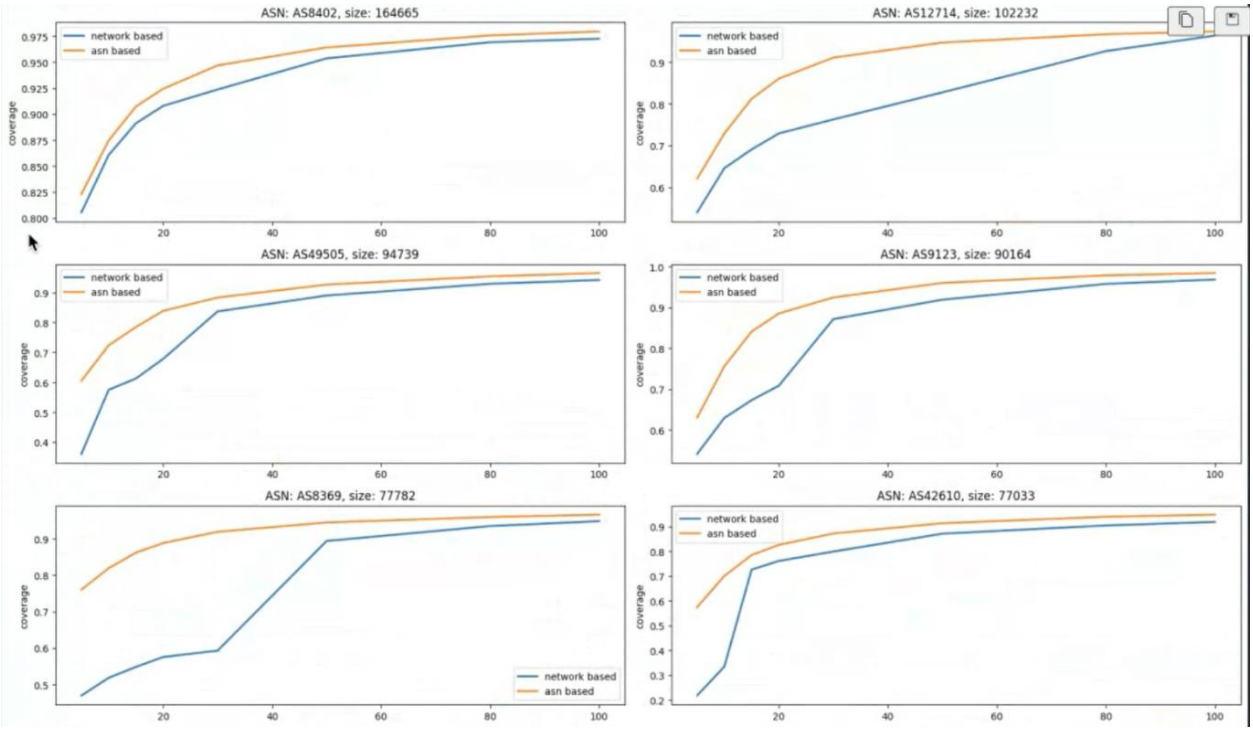

Рис.4 Сравнение кривых метода с предсказанием (желтая) и полного перебора портов (синяя), которые разбиты по ASN-ам

Тут мы видим графики зависимости количества сканируемых портов и приближения к эталону. Верхняя кривая отвечает за статистику, подведенную по группе узлов, нижняя – по всем найденным IP-адресам. Можно заметить, насколько быстрее поднимается верхняя кривая. Данный факт говорит о том, что для некоторых сегментов (в данном случае ASN-ов) можно построить профиль наиболее важных для сканирования портов, которые покроют большую часть сервисов.

Для того, чтобы нам получить статистику, в идеале необходимо провести полное сканирование портов по 65535 портам, что для всего сегмента также является дорогим удовольствием.

www.cyberok.ru
info@cyberok.ru
+7 (495) 137-7337

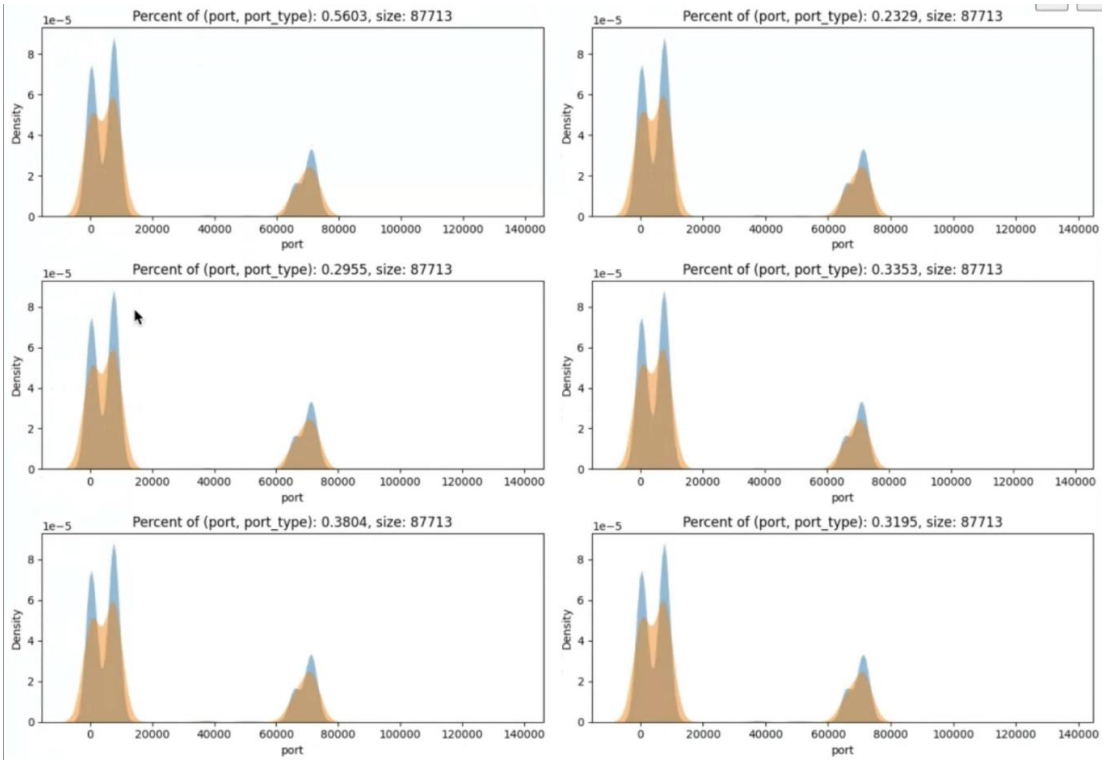

Рис.5 Сравнение графиков сканирования всех IP-адресов сегмента (синий) и 10% IP-адресов сегмента (желтый)

На рисунке 5 показано покрытие сетевых портов и сервисов при взятии 10% IP-адресов сегмента. Можно заметить, насколько они совпадают, что подтверждает гипотезу о возможности построения профиля по некоторой части данных.

Метод 2: Автоматизация определения программного обеспечения

Если с сетевым сканированием все проще – это просто ресурсы машин, то с написанием правил определения программного обеспечения все сложнее – над этим трудятся наши эксперты. Одной из фаз является пассивное определение, в котором мы используем регулярные выражения. При этом процесс содержит в себе 2 подхода. Первый – использование правил, написанных экспертами, второй – правила, полученные в автоматизированном режиме. При автоматизации используется большая языковая модель, которая натренирована на ответах сервисов, как показано на рисунках 6-7-8. В результате получаем некоторую регулярку с меньшей точностью определения ПО, но при этом сильно экономим ресурсы.
Я, к примеру, никогда не слышал про Ratchet, а вы?

www.cyberok.ru
info@cyberok.ru
+7 (495) 137-7337


```
"is_processed": false,
"num_of_matches": 177,
"re": "X-Powered-By: \\ Ratchet/0\\.4\\..*",
"re_match": "Ratchet/0.4.1",
"readability_test": {
    "flesch_reading": false,
    "is_first_upper": true,
    "ner": true,
    "num_is_alpha": true,
    "punct_ratio": false,
    "tokens_ratio": 10.0
},
"sample": [
    "HTTP/1.1 426 Upgrade header MUST be provided\r\nConnection:
    "HTTP/1.1 426 Upgrade header MUST be provided\r\nConnection:
    "HTTP/1.1 426 Upgrade header MUST be provided\r\nConnection:
    "HTTP/1.1 426 Upgrade header MUST be provided\r\nConnection:
    "HTTP/1.1 426 Upgrade header MUST be provided\r\nConnection:
]
```


Рис.6 Пример полученного регулярного выражения

```
0. Server: Synapse/1.96.1 is_processed=False Matches: 56
1. Server: Synapse/1.90.0 is_processed=False Matches: 8
2. Server: JAWS/1.0 is_processed=False Matches: 32
3. Server: Werkzeug/3.0.1 Python/3.12.1 is_processed=False Matches: 77
4. Server: Nimble/3.7.11-9 is_processed=False Matches: 51
5. X-Powered-By: Ratchet/0.4.1 is_processed=False Matches: 177
6. Server: IPC/2.0.0 is_processed=False Matches: 466
7. Server: Brovotech/2.0.0 is_processed=False Matches: 212
8. Server: Airee/Cloud is_processed=False Matches: 251
9. Server: BaseHTTP/0.6 Python/3.10.7 is_processed=True Matches: 38
10. Server: Microsoft-IIS/10.0 is_processed=False Matches: 97
11. X-Powered-By: ASP.NET is_processed=True Matches: 71
12. Server: WSGIServer/0.2 CPython/3.8.10 is_processed=False Matches: 46
```

Рис.7 Примеры определенного программного обеспечения


```
-Version: 13\r\nUpgrade: websocket\r\nX-Powered-By: Ratchet/0.4.4",
-Version: 13\r\nUpgrade: websocket\r\nX-Powered-By: Ratchet/0.4.4",
-Version: 13\r\nUpgrade: websocket\r\nX-Powered-By: Ratchet/0.4.4",
-Version: 13\r\nUpgrade: websocket\r\nX-Powered-By: Ratchet/0.4.4",
-Version: 13\r\nUpgrade: websocket\r\nX-Powered-By: Ratchet/0.4.4"
```


Рис.8 Пример кластеризации ресурсов, на основании которого формируется регулярное выражение

www.cyberok.ru
info@cyberok.ru
+7 (495) 137-7337

Когда наконец все данные собрали и сложности преодолели, приступаем к охоте.

Особенности национальной охоты: Тактики и техники атак

Мы выделили 5 тактик и техник, которые могут быть использованы злоумышленниками при планировании атак с учетом региональных особенностей, потому что не все так однозначно:

1. Мониторинг трендовых уязвимостей
2. Типовое для региона программное обеспечение
3. Зачастую уязвимые технологии
4. Порты с характерным ПО
5. То, что пытаются скрыть

Разберем каждый из них на примерах.

1. Мониторинг трендовых уязвимостей

Первая тактика, которая может быть нами использована – мониторинг трендовых уязвимостей. Из последнего:

- TeamCity (CVE-2024-27198)
- ConnectWise (CVE-2024-1709)
- Fortinet (CVE-2024-21762)
- Wordpress (...)
- Gitlab (CVE-2023-7028)

Важно отметить, что помимо стандартных метрик, так как CVSS, важно учитывать и распространённость программного обеспечения. Так, к примеру, уязвимость в ConnectWise, хоть и массово разошлась в информационном поле, но практически не имела влияния на российский сегмент Интернета.

В итоге данная тактика может быть представлена режимом «ждуна», в котором добавление новых проверок в известном ПО происходит довольно быстро и иногда не требует пересканирования, что означает практически моментальный результат.

Рис.9 Ждун вместе с горожанами ожидает прихода широкополосного Интернета в Норильск. А мы ждуним классных вулнов.

Ниже приведена статистика, показывающая, что разные уязвимости оказывают разное влияние и на количество ресурсов, и на потенциальное количество затрагиваемых организаций. Можно сделать вывод, что выход новой критичной уязвимости в GitLab может быть фатален. Про то, как исправляются подобного рода системы, поговорим чуть позднее.

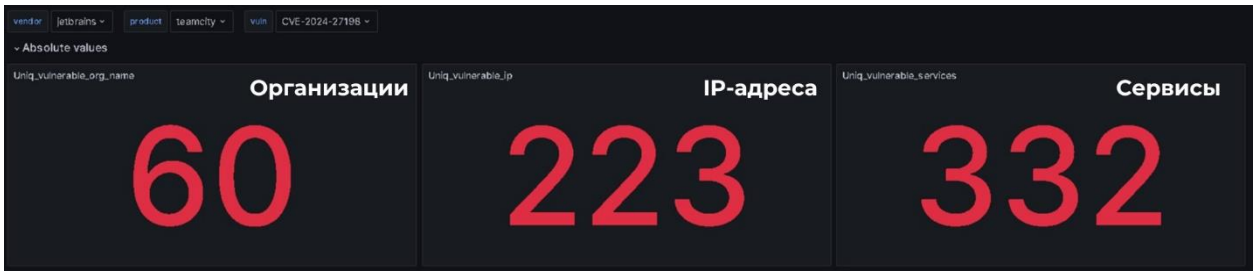

Рис.10 TeamCity (CVE-2024-27198)

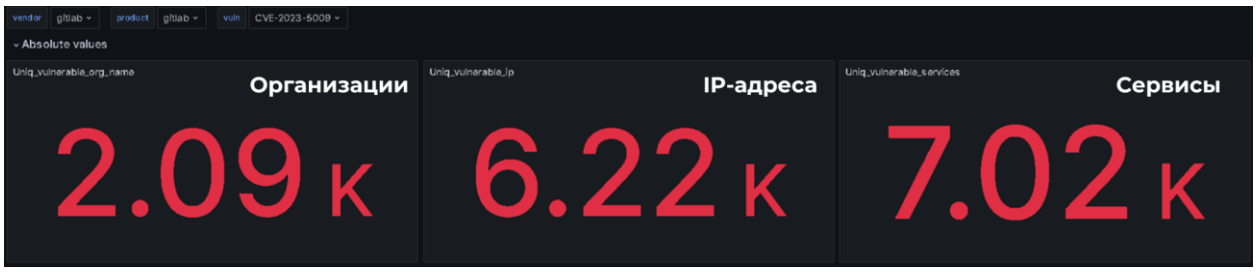

Рис.11 GitLab (CVE-2023-5009)

2. Типовое для региона ПО

Довольно лакомой целью является типовое для региона программное обеспечение. Вот несколько примеров:

- 1C-Bitrix (уязвимости, мисконфиги, следы эксплуатации);
- Trueconf (CVE-2022-46764);
- Terrasoft/Creatio (BDU-2023-04519);
- Websoft HCM (BDU-2022-06939).

Тут начинает сказываться региональная специфика. Для некоторых уязвимостей нет CVE, и соответственно начинают ломаться инструменты, завязанные на списки CVE. Например, метрика EPSS, основная суть которой заключается в цитируемости уязвимости в международном информационном поле. Для нас критичная уязвимость в 1C-Bitrix – это ужасно, а остальные о ней не слышали.

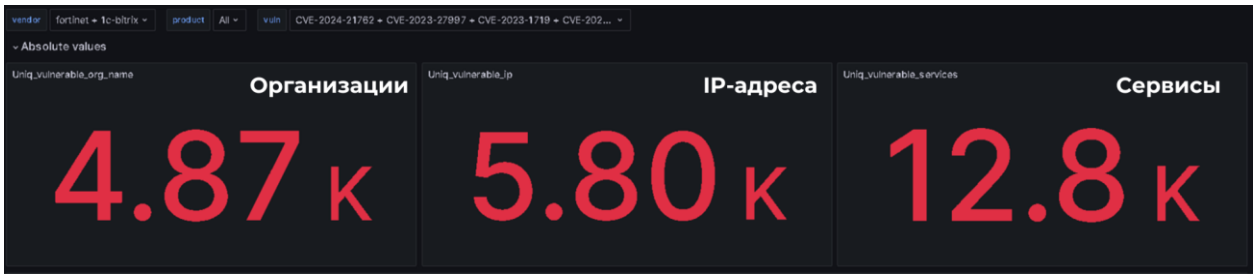

Рис.12 Уязвимость 1C-Bitrix

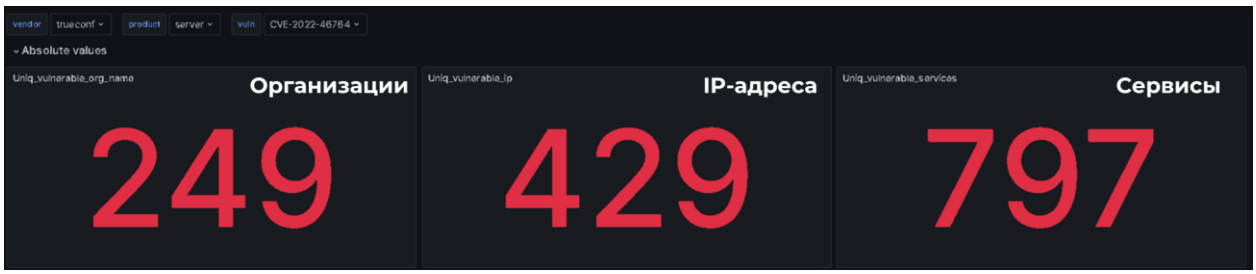

Рис.13 Уязвимость Trueconf (CVE-2022-46764)

Если критичность нахождения уязвимостей в 1C-Bitrix для большинства понятна, так как в мае прошлого года очень многих затронула череда взломов и последующей постэксплуатации, то с TrueConf менее прозрачная ситуация. Скорее всего, это был тот класс ПО, на который никто из злоумышленников не обращал внимание, но при этом уязвимость в нем содержалась. Ситуация усугублялась еще и тем, что данное программное обеспечение используется во множестве серьезных организаций.

Стоит также отметить, что несмотря на большое количество найденных уязвимых узлов, каждый из вендоров оперативно уведомлял своих клиентов, что сказалось на статистике устранения угроз. Об этом поговорим далее.

3. Зачастую уязвимые технологии

Дополнительным и, возможно, менее очевидным, но от этого не менее массовым, являются атаки на второстепенные ресурсы организаций, такие как веб-камеры, различные хранилища. Очень часто данные устройства практически не обновляются, в результате чего имеют очень старые уязвимости. Мы называем их CVT (Common Vulnerable Technologies).

Приведем примеры таких устройств, упорядоченных по распространённости. В некоторых также указано количество уязвимых устройств (через /):

- Mikrotik (CVE-2018-14847) – 94000+;
- Hikvision (CVE-2017-7921, CVE-2021-36260) - 3000/75000+;
- Dahua (CVE-2021-33044) - 10000+;
- Xiongmai (CVE-2017-7577, CVE-2018-10088) - 900/10000+.

4. Порты с характерным ПО

Когда большинство распространённых низко висящих фруктов сорвано, можно переходить к поиску аномалий на ранее неизвестных портах. Рассмотрим следующие порты с аномалией:

- 1801 – msmq (CVE-2023-21554);
- 4307 – TrueConf (CVE-2022-46764)
- 10050 – Zabbix agent
- 7547 - ?

Первая из аномалий была обнаружена, благодаря уязвимости в msmq 2023 года. В ходе сравнения результатов СКИПА с Shodan и Censys было обнаружено, что данный порт не сканируется конкурентами. Стоит отметить, что через некоторое время и они обратили внимание на этот протокол и, как следствие, на порт. Похожим примером является практически отсутствующее определение ПО TrueConf на порту 4307.

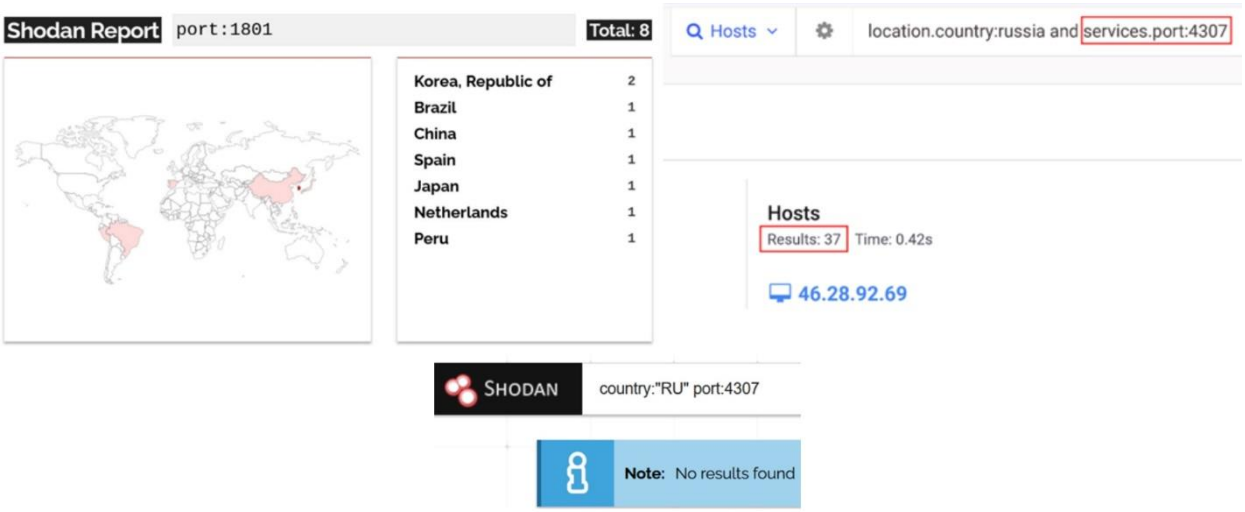

Рис.14 Статистика распространённости портов 1801 и 4307 в Shodan и Censys

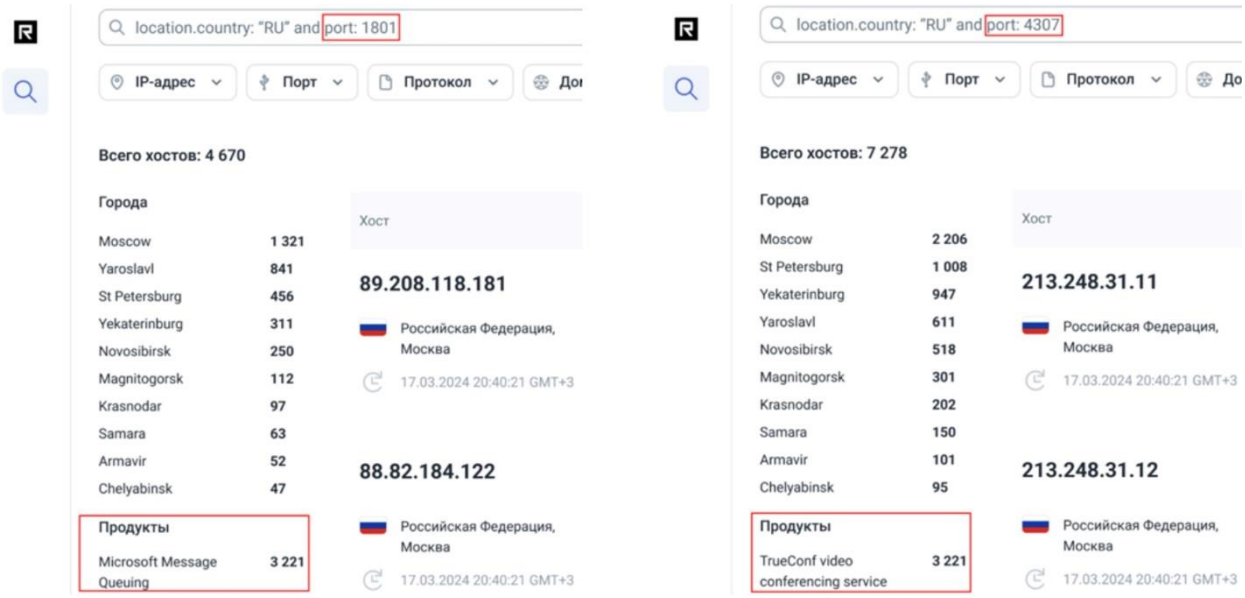

Рис.15 Статистика распространённости по портам 1801 и 4307 в СКИПА

Но что же скрывается за портом 7547?

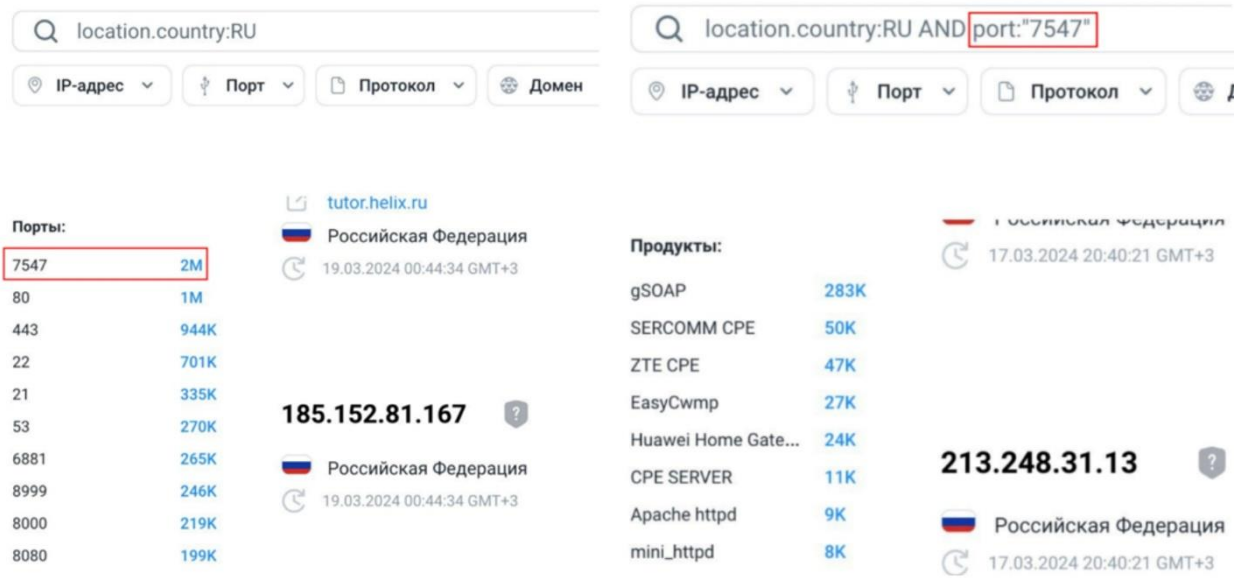

Рис.16 Статистика распространённости порта 7547 и продуктов на нем в СКИПА

Возможно, вы удивитесь (мы вот удивились), но порт 7547 является самым распространённым в Рунет. При помощи технологии СКИПА была подведена статистика не только по распространённости этого порта, но и по продуктам, которые на нем располагаются. Преимущественно они связаны с устройствами класса CPE (Customer Premises Equipment) или абонентским оборудованием. Обычно это типовые устройства, предоставляемые провайдерами, и работающие по протоколу CWMP (CPE WAN Management Protocol) (в свою очередь работает на базе HTTP и SOAP). Так как

данные устройства являются типовыми, как и их настройка, то любые потенциально найденные уязвимости или мисконфигурации ставят под угрозу сотни тысяч устройств.

5. То, что пытаются скрыть

Если всех перечисленных выше тактик и техник показалось недостаточно, то свой взор можно обратить на сервисы, которые пользователи Интернета «пытаются» скрыть. Обычно это добавление цифры к стандартному для данного протокола или ПО порту или смена порта на нетипичный в верхнем диапазоне. Для злоумышленника обычно это красный флажок, говорящий о том, что там можно и музыку послушать, и телевизор посмотреть, и зайти туда, где вас возможно не ждут.

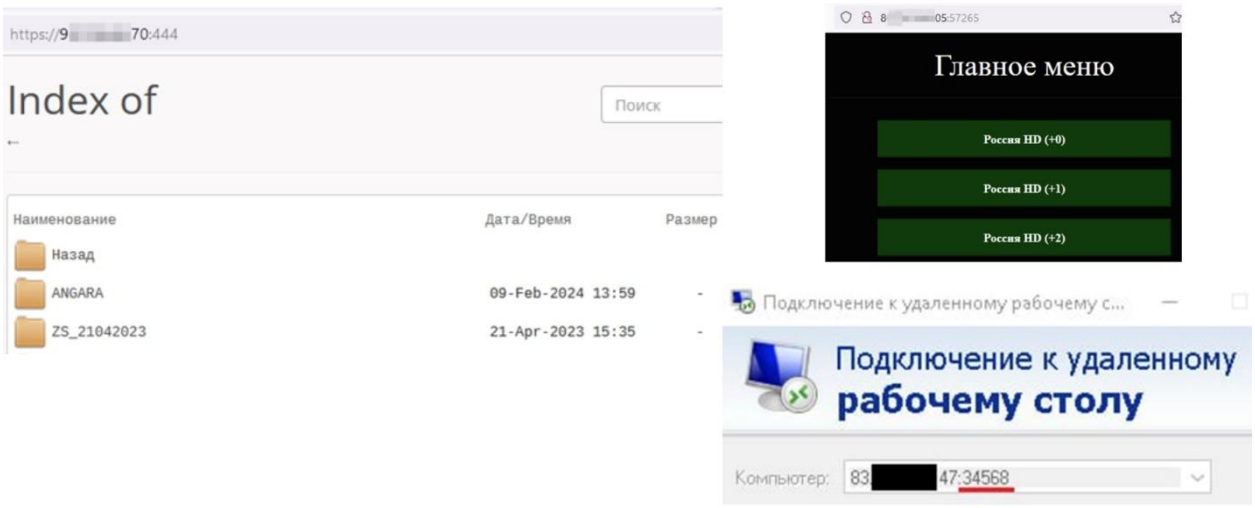

Рис.17 Немного примеров ресурсов, которые «пытаются» скрыть

Результаты

Итак, мы проверили множество подходов, нашли огромное количество уязвимостей, но возникает опрос – что же с этим делать?

Поскольку в нашей практике встречаются различные реакции владельцев ресурсов от: «Немедленно прекратите, я обращаюсь в правоохранительные органы!» до: «Ребята, давайте взаимодействовать вместе для устранения уязвимых сервисов!», то хотелось бы поделиться теми организационными и техническими выводами, к которым мы пришли и подкрепить это статистикой.

Трендовые уязвимости: Fortinet

Самым типичным случаем является волнообразное исправление программного обеспечения. На графике ниже представлена зависимость количества уязвимых сервисов от времени в продукте Fortinet (fortios). Пики – это выход новых уязвимостей, между которыми наблюдаются спады – периоды их исправления.

www.cyberok.ru
info@cyberok.ru
+7 (495) 137-7337

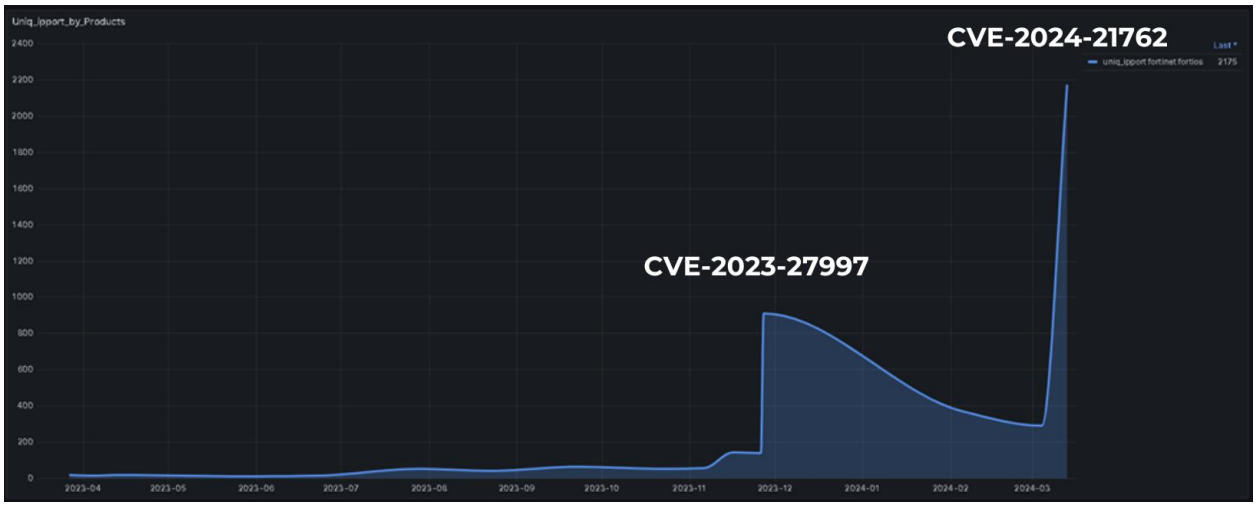

Рис.18 Динамика уязвимых сервисов продукта Fortinet(fortios)

Трендовые уязвимости: Atlassian Confluence

Также довольно типичным поведением, которое мы можем проследить на основании продукта Atlassian Confluence, является оперативное исправление продуктов с последующей стагнацией.

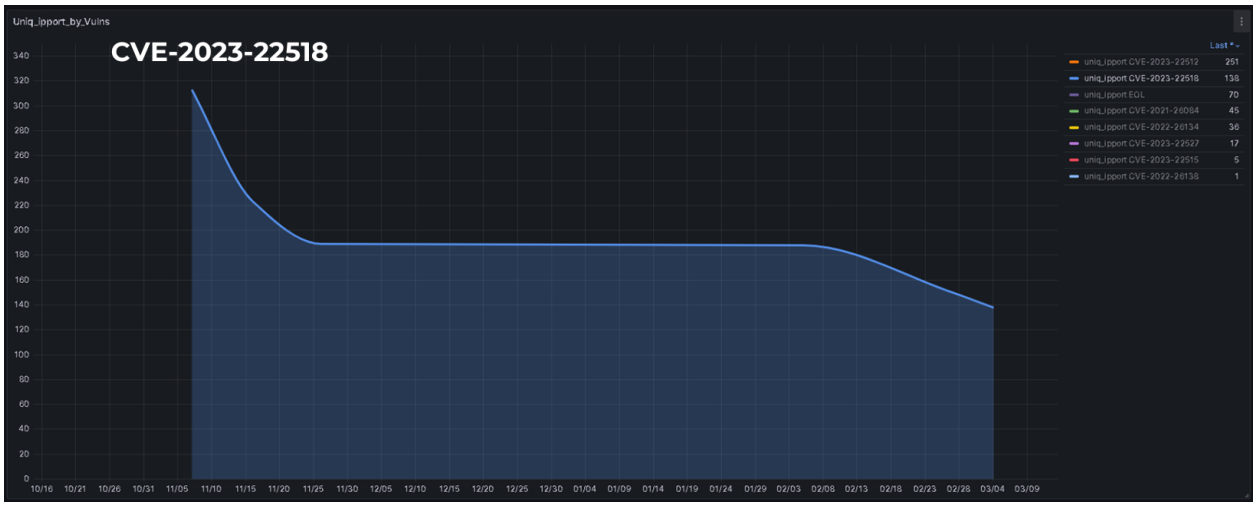

Рис.19 Динамика уязвимых сервисов продукта Atlassian Confluence

Компрометация, уязвимости, бэкдоры в CMS 1C-Bitrix

Более продуктивным как с точки зрения организационных мер, так и с технических было взаимодействие по продукту 1C-Bitrix управление сайтом.

Рис.20 Динамика скомпрометированных и уязвимых сервисов продукта 1C-Bitrix

В ходе мониторинга, при помощи технологии СКИПА, в конце мая предыдущего года была обнаружена кампания, направленная на подмену содержимого множества ресурсов однотипным контентом – это первый пик, отмеченный желтым цветом. Далее при взаимодействии с коллегами из компании 1C-Bitrix нами были обнаружены первые индикаторы компрометации, выраженные в виде размещенных на зараженных сайтах бэкдоров – это второй пик, отмеченный синим цветом. Далее в результате нашего полного сканирования было обнаружено более 8 тысяч уязвимых ресурсов (уязвимости в плагинах vote и htmleditor). Если сравнить соотношение уязвимых (зеленый график) к ресурсам с подтвержденным бэкдором (синий график), то можно оценить количество сервисов, по которым получилось отреагировать до закрепления злоумышленников. Помимо этого, велось плотное взаимодействие с хостерами для оповещения клиентов. Одним из последних возможных побуждений для самых безответственных к обновлению ресурсов было применение ТСПУ, которое дало свои плоды.
Второй пик подтвержденных бэкдоров, которые вы можете заметить – это непрекращающееся взаимодействие с вендором, который был сильно заинтересован в исправлении как можно большого количества своих клиентов.

Работа по данному продукту является показательным применением упорядоченного комплекса организационных и технических мер, который в итоге привел к успешной локализации и исправлению угрозы.

Типовое для региона ПО: TrueConf

Похожим примером является взаимодействие с компанией TrueConf. На рисунке ниже изображен график зависимости количества уязвимых сервисов от времени.

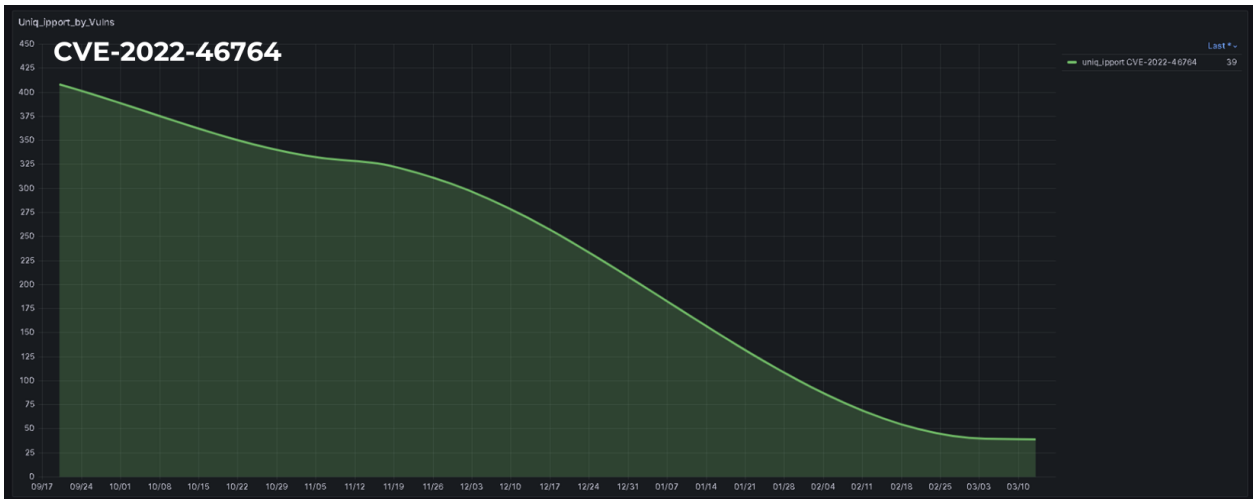

Рис.20 Динамика уязвимых сервисов продукта TrueConf

В ходе взаимодействия с компанией SolidLab, а также при помощи технологии СКИПА, были собраны все уязвимые сервисы, а также потенциально уязвимые версии программного обеспечения. Далее вся информация была передана вендору и по динамике обнаружения мы заметили значительное снижение ресурсов, находящихся под угрозой.

Выводы

В итоге, проведенное исследование позволяет нам сделать следующие выводы, которые наверняка все знают, но очень важно про них напомнить:

1. Следите за новыми уязвимостями

2. Знайте свой периметр, чтобы принять решение о важности той или иной угрозы конкретно для вас

3. Оперативно действуйте в случае обнаружения уязвимости на своем периметре

4. Старайтесь не только скрываться, но и становиться безопаснее

Автор: Игорь Первушин – руководитель направления создания базы знаний компании CyberOK.