

# Тема №8: Безопасность серверов и проектов. Опыт Southbridge

## Тема №3: Работайте как Southbridge. Технический регламент

- Защита SSH. Использование OTP для доступа к серверам. Скрипты и их применение
- Защита сайта от DDoS-атак
- Скрипты защиты от http-флуда, сканирования портов, проверки целостности пакетов
- Скрипты проверки ПО на уязвимости и автоматическое обновление

РЕД  
СЛЁРМ

+

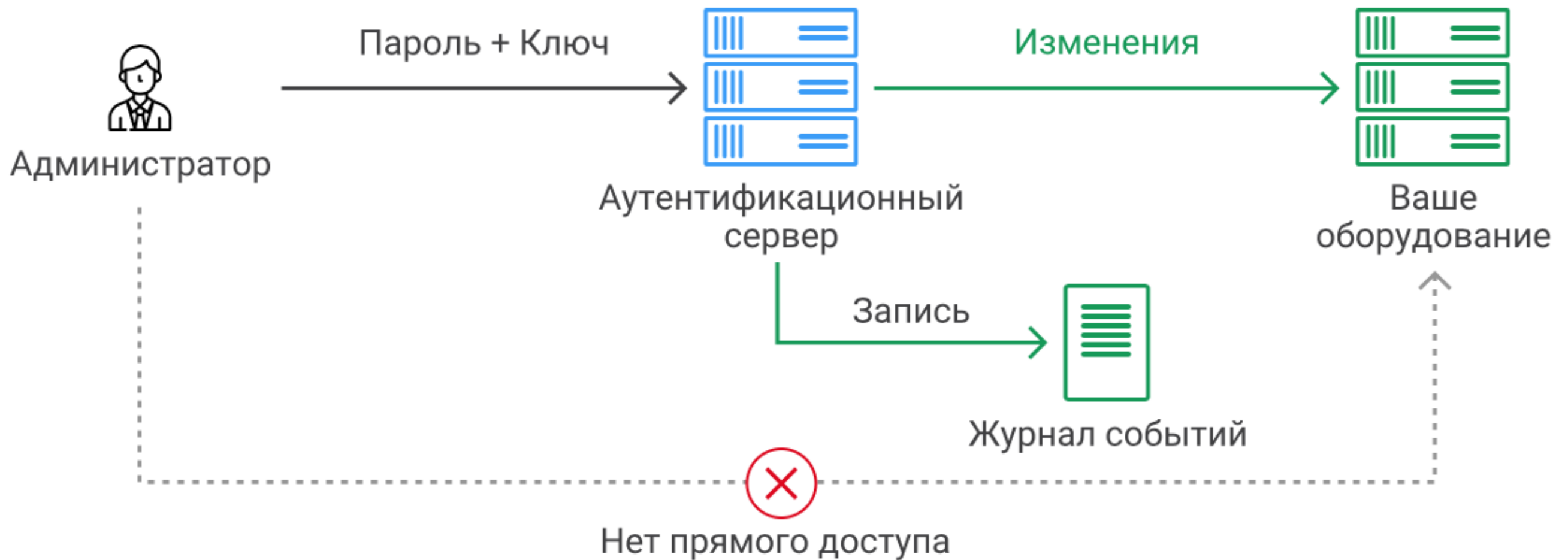


Southbridge



# Защита SSH. Использование OTP для доступа к серверам. Скрипты и их применение

# Организация доступа



Google Authenticator: <https://github.com/google/google-authenticator-libpam>.

Auto ssh-agent: [https://gitlab.slurm.io/red/slurm/blob/master/practice/8.ssh-agent/bash\\_profile](https://gitlab.slurm.io/red/slurm/blob/master/practice/8.ssh-agent/bash_profile)



# Организация доступа

- <https://gitlab.slurm.io/red/slurm/blob/master/practice/8.iptables/iptables.ssh>
- <https://gitlab.slurm.io/red/slurm/blob/master/practice/8.iptables/ssh.iptables.cfg>
- /etc/ssh.iptables.local.cfg
- Sudo user check  
[https://gitlab.slurm.io/red/slurm/blob/master/practice/8.sudo-user-check/\\_sb\\_sudo\\_user\\_check.sh](https://gitlab.slurm.io/red/slurm/blob/master/practice/8.sudo-user-check/_sb_sudo_user_check.sh)

РЕД  
СЛЁРМ

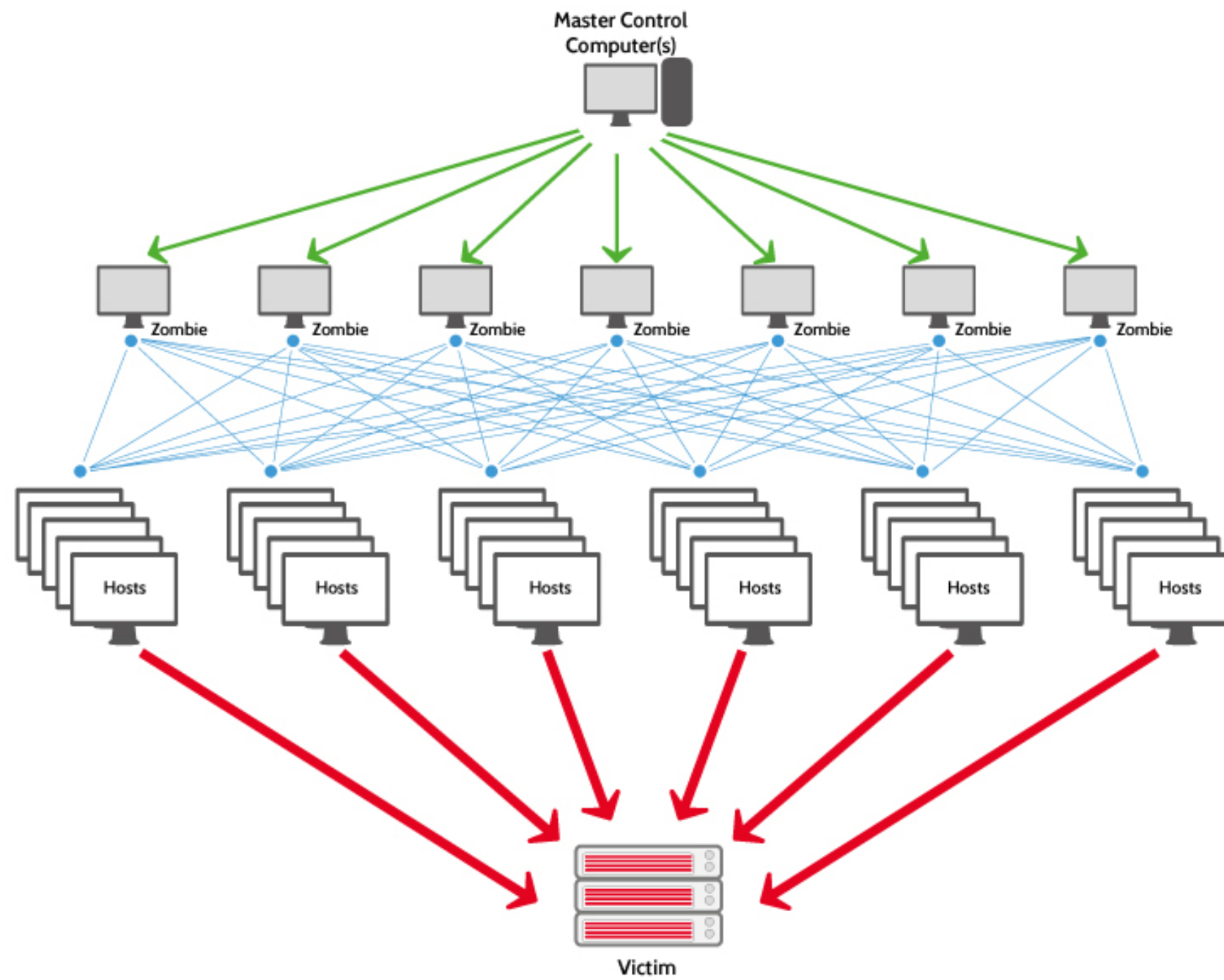
+

  
Southbridge

# Защита сайта от DDoS-атак

slurm.io

# DDoS



# ■ Типы DDoS атак

- Полосы пропускания
- Исчерпание системы
- Зацикливание
- Ложные атаки
- HTTP протокол
- Смурф-атака
- UDP-флуд
- SYN-флуд
- Тяжёлые пакеты
- Лог-файлы
- Программный код



# Защита от DDoS-атак

Использовать модуль testcookie

<https://github.com/kyprizel/testcookie-nginx-module>

Пример настройки:

<https://github.com/kyprizel/testcookie-nginx-module#example-configuration>

# Защита от DDoS-атак

Код 404

```
location /search {  
    return 444;  
}
```

```
ipset -N ban iphash  
tail -f access.log | while read LINE; do echo "$LINE" | \  
    cut -d'"' -f3 | cut -d' ' -f2 | grep -q 444 && ipset -A  
    ban "${L%% *}"; done
```

# Защита от DDoS-атак

## Баним по геопризнаку

1. Подключите к nginx GeoIP-модуль ([wiki.nginx.org/HttpGeoipModule](http://wiki.nginx.org/HttpGeoipModule)).
2. Выведите информацию о геопривязке в access log.
3. Далее, модифицировав приведенный выше шелл-скрипт, прогнрайте accesslog nginx'а и добавьте отфутболенных по географическому признаку клиентов в бан.

# Диагностика проблемы

1. Юзайте профайлер и отладчик
2. Анализируйте ошибки (request\_time, upstream\_response\_time)
3. Отслеживайте количество запросов в секунду

```
echo $(( $(fgrep -c "$(env LC_ALL=C date --date=@$(($(date \\\n    +%s)-60)) +%d/%b/%Y:%H:%M)" "$ACCESS_LOG")/60))
```



# Тюнинг веб-сервера

Лимитируем ресурсы (размеры буферов) в nginx

- client\_header\_buffer\_size\_\_
- large\_client\_header\_buffers
- client\_body\_buffer\_size
- client\_max\_body\_size

# Тюнинг веб-сервера

Настраиваем тайм-ауты в nginx

- `reset_timeout_connection on;`
- `client_header_timeout`
- `client_body_timeout`
- `keepalive_timeout`
- `send_timeout`

1. Выставляем математически минимальное значение параметра.
2. Запускаем прогон тестов сайта.
3. Если весь функционал сайта работает без проблем — параметр определен. Если нет — увеличиваем значение параметра и переходим к п. 2.
4. Если значение параметра превысило даже значение по умолчанию — это повод для обсуждения в команде разработчиков.

# Готовим ОС

## Тюним ядро

- net.ipv4.tcp\_fin\_timeout
- net.ipv4.tcp\_{r,w}mem
- net.core.{r,w}mem\_max

```
sysctl -w net.core.rmem_max=8388608
sysctl -w net.core.wmem_max=8388608
sysctl -w net.ipv4.tcp_rmem='4096 87380 8388608'
sysctl -w net.ipv4.tcp_wmem='4096 65536 8388608'
sysctl -w net.ipv4.tcp_fin_timeout=10
```

Ревизия /proc/sys/net/\*\*

# ■ Провайдеры защиты от DDoS



StormWall<sub>PRO</sub><sup>TM</sup>





РЕД  
СЛЁРМ

+



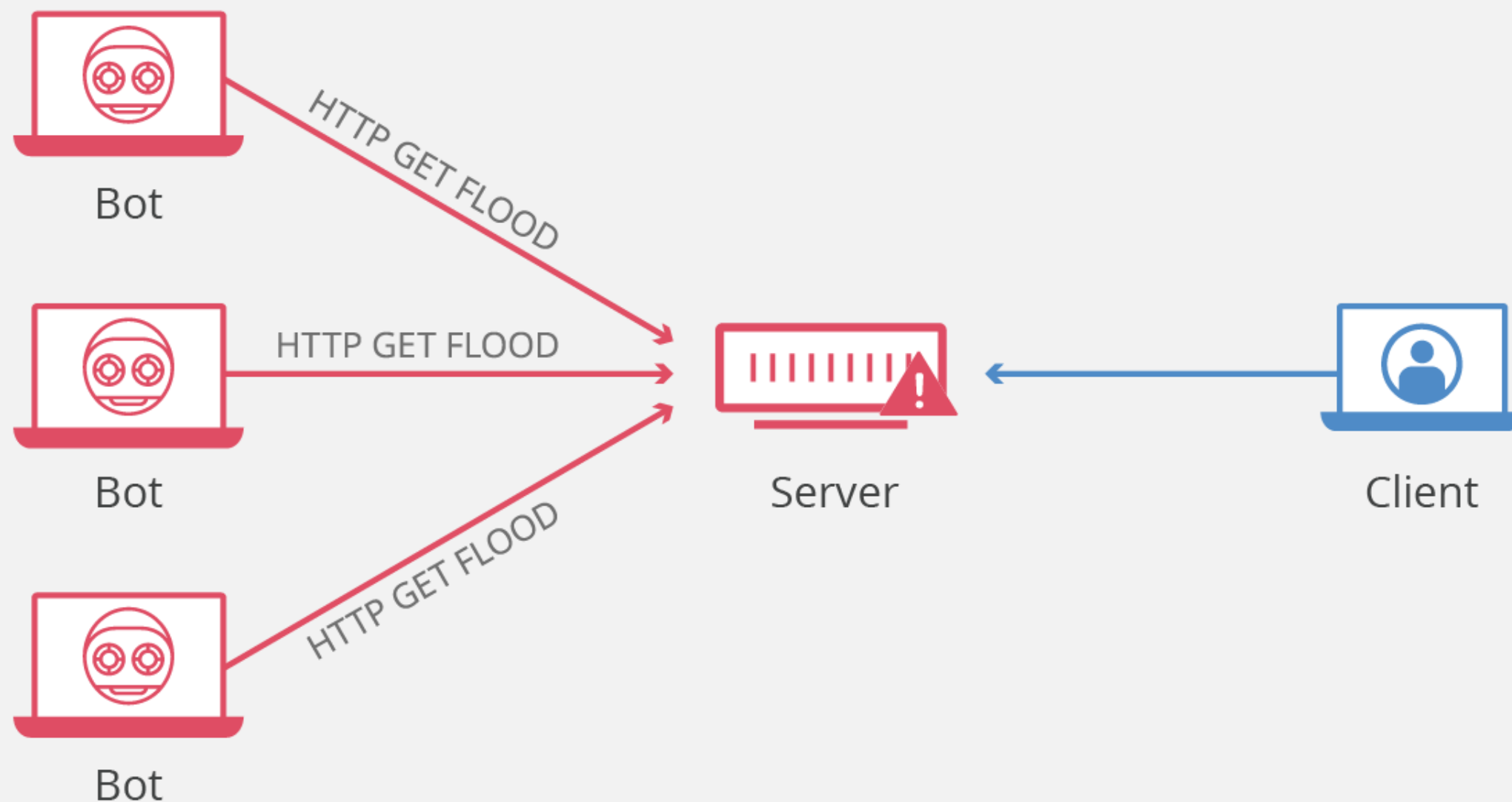
Southbridge



# Скрипты защиты от http- флуда, сканирования портов, проверки целостности пакетов

# Защита от HTTP-флуда


## HTTP Flood Attack



<https://gitlab.slurm.io/red/slurm/tree/master/practice/8.flood-protect>

# Сканирование портов

[https://gitlab.slurm.io/red/slurm/blob/master/practice/8.check-open-ports/check\\_openports.pl](https://gitlab.slurm.io/red/slurm/blob/master/practice/8.check-open-ports/check_openports.pl)



```
Starting Nmap 6.00 ( http://nmap.org ) at 2012-05-17 12
Nmap scan report for scanme.nmap.org (74.207.244.221)
Host is up (0.00031s latency).
Not shown: 997 closed ports
PORT      STATE SERVICE      VERSION
22/tcp    open  ssh          OpenSSH 3p1 Debian 3ubuntu7
| ssh-hostkey: 1024 10:0a:d6:67:54:9d:00:00:00:00:00:00:00:00:00:00
|_ 2048 79:f8:00:00:00:00:00:00:00:00:00:00:00:00:00:00
80/tcp    open  http         Apache/2.2.8 ((Ubuntu))
|_ http-ti
9929/tcp  open
Device type: general purpose
Running: Linux 2.6.X|3.X
OS CPE: cpe:/o:linux:kernel:2.6 cpe:/o:linux:kernel:3
OS details: Linux 2.6.32 - 2.6.39, Linux 2.6.38 - 3.0
Network Distance: 2 hops
Service Info: OS: Linux; CPE: cpe:/o:linux:kernel
```

# Проверка целостности пакетов

<https://gitlab.slurm.io/red/slurm/tree/master/practice/8.fix-rpm>

```
S.5....T.  c /etc/systemd/journald.conf
```

**5** – контрольная сумма MD5  
**S** – размер  
**L** – символическая ссылка  
**T** – дата изменения файла  
**D** – устройство  
**U** – пользователь  
**G** – группа  
**M** – режим (включая разрешения и тип файла)  
**?** – файл не удалось прочитать





# Проверка целостности пакетов

## Файл конфигурации

- `/srv/southbridge/etc/fix-rpm.conf.dist` - файл конфигурации, устанавливаемый со скриптом по умолчанию;
- `/srv/southbridge/etc/fix-rpm.conf` - файл конфигурации для внесений изменений через слак;
- `/srv/southbridge/etc/fix-rpm.local.conf` - файл конфигурации для локальных изменений.



РЕД  
СЛЁРМ

+

  
Southbridge

# Скрипты проверки ПО на уязвимости и автоматическое обновление

## Скрипты проверки ПО на уязвимости и автоматическое обновление

<https://github.com/videns/vulners-scanner>

```
# ./linuxScanner.py
=====
Host info - Host machine
OS Name - Darwin, OS Version - 15.6.0
Total found packages: 0
=====
Host info - docker container "java:8-jre"
OS Name - debian, OS Version - 8
Total found packages: 166
Vulnerable packages:
  libgcrypt20 1.6.3-2+deb8u1 amd64
    DSA-3650 - 'libgcrypt20 -- security update', cvss.score - 0.0
  libexpat1 2.1.0-6+deb8u2 amd64
    DSA-3597 - 'expat -- security update', cvss.score - 7.8
  perl-base 5.20.2-3+deb8u4 amd64
    DSA-3628 - 'perl -- security update', cvss.score - 0.0
  gnupg 1.4.18-7+deb8u1 amd64
    DSA-3649 - 'gnupg -- security update', cvss.score - 0.0
  gpgv 1.4.18-7+deb8u1 amd64
    DSA-3649 - 'gnupg -- security update', cvss.score - 0.0
```

## Скрипты проверки ПО на уязвимости и автоматическое обновление

[https://gitlab.slurm.io/red/slurm/blob/master/practice/8.yum-security-update/sb\\_yum\\_security\\_check.sh](https://gitlab.slurm.io/red/slurm/blob/master/practice/8.yum-security-update/sb_yum_security_check.sh)

```
[root@vcptest ~]
[root@vcptest ~] yum updateinfo list
Loaded plugins: amazon-id, rhui-lb, search-disabled-repos
RHBA-2016:0547 bugfix      NetworkManager-1:1.0.6-29.el7_2.x86_64
RHBA-2016:1285 bugfix      NetworkManager-1:1.0.6-30.el7_2.x86_64
RHBA-2016:0547 bugfix      NetworkManager-config-server-1:1.0.6-29.el7_2.x86_64
RHBA-2016:1285 bugfix      NetworkManager-config-server-1:1.0.6-30.el7_2.x86_64
RHBA-2016:0547 bugfix      NetworkManager-libnm-1:1.0.6-29.el7_2.x86_64
RHBA-2016:1285 bugfix      NetworkManager-libnm-1:1.0.6-30.el7_2.x86_64
RHBA-2016:0547 bugfix      NetworkManager-team-1:1.0.6-29.el7_2.x86_64
RHBA-2016:1285 bugfix      NetworkManager-team-1:1.0.6-30.el7_2.x86_64
RHBA-2016:0547 bugfix      NetworkManager-tui-1:1.0.6-29.el7_2.x86_64
RHBA-2016:1285 bugfix      NetworkManager-tui-1:1.0.6-30.el7_2.x86_64
RHBA-2016:0183 bugfix      avahi-autoipd-0.6.31-15.el7_2.1.x86_64
RHBA-2016:0183 bugfix      avahi-libs-0.6.31-15.el7_2.1.x86_64
RHBA-2016:1522 bugfix      bash-4.2.46-20.el7_2.x86_64
RHSA-2016:0459 Important/Sec. bind-libs-lite-32:9.9.4-29.el7_2.3.x86_64
RHSA-2016:0459 Important/Sec. bind-license-32:9.9.4-29.el7_2.3.noarch
RHEA-2016:0182 enhancement  ca-certificates-2015.2.6-70.1.el7_2.noarch
```



# Вопросы?

