

## Into The Core

IN-DEPTH EXPLORATION OF WINDOWS 10 IOT CORE



X-Force Advanced Research IBM Security



### Agenda

Introduction

Internals

**Attack Surface** 

Hacking

Recommendations

Conclusion

## Introduction

### Overview

Edition	Description	Target Devices
Windows 10 IoT Enterprise	UWP apps, Win32 apps, desktop shell, x86, advanced lockdown	Kiosk, POS, ATM, Medical devices
Windows 10 IoT Mobile	UWP apps, multiuser support, lockdown features	Mobile POS, Industry hand held terminals
Windows 10 IoT Core	For low-cost, low-power devices. UWP apps only. ARM and x86	Smart home devices, IoT gateway, digital signage

### Overview

### Raspberry Pi 2 & 3

- ARM
- 32-bit
- On-board Wi-fi and Bluetooth, Ethernet
- 4 x USB 2.0



### Minnowboard Max

- x86
- 32-bit
- Ethernet
- 1 x USB 2.0, 1 x USB 3.0



### Dragonboard 410c

- ARM
- · On-board Wi-fi and Bluetooth
- 2 x USB 2.0



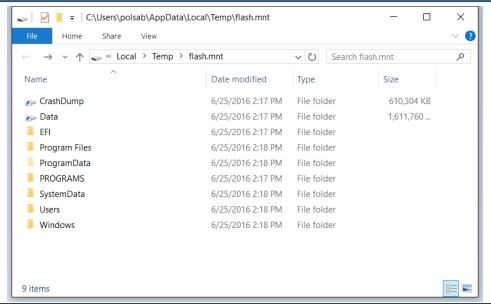
## Internals

### Internals > FFU

```
C:\>ImgMount.exe "c:\Program Files (x86)\Microsoft IoT\FFU\MinnowBoardMax\flash.ffu"

WP8 ROM Image Tools v.1.0.204
htc ROM Image Editor (-) 2007-2012 AnDim & XDA-Developers
ImgMount Tool v.1.0.15

(htcRIE) Mounting the image file: 'c:\Program Files (x86)\Microsoft IoT\FFU\MinnowBoardMax\flash.ffu'
Loading .FFU image ... ok
Creating virtual disk ... ok
Mounting MainOS partition as: '\\flash.mnt\' ... ok
(htcRIE) Successfully mounted an image file.
```



### Internals > Partition Layout

Partition	File System	Mount Point	Contents
EFI System Partition	FAT	C:\EFIESP	Boot manager, boot configurations, UEFI applications
Crash dump partition	FAT32	D:	Crash dump data
Main OS	NTFS	C:	OS, registry hives, OEM applications
Data partition	NTFS	U:	Applications, application data, user data

IBM Security

### Internals > Boot Process

Device powers on and runs SoC firmware bootloader

Bootloader launched the UEFI environment and UEFI applications

UEFI environment launches Boot Manager (C: \EFIESP\EFI\Microsoft\boot\bootmgfw.efi)

Boot Manager launches Windows Boot Loader (C:\Windows\System32\Boot\winload.efi)

Windows Boot Loader launches main OS

### Internals > Apps

- Universal Windows Platform (UWP) apps
  - Foreground/default apps
  - Background app
- Console applications
  - Win32 apps
  - No UI
  - C++ only

- Headed/Headless mode
  - Ul or no Ul

### Internals > Security

Windows Def., der Microsoft Pass ort

Virtualization
Based Security
(VBS)

**Device Guard** 

**Credential Guard** 

Hypervisor Code Integrity (HVCI)

### Internals > Security

Address
Space Layout
Randomization
(ASLR)

Data
Execution
Prevention
(DEP)

Control Flow Guard (CFG)

### Internals > Security

Туре	Description
Firmware TPM	TPM implemented in the SoC
Discrete TPM	Chip module that can be attached to a board
Software TPM	Software emulated TPM used in development

### Secure Boot

- Prevents device tampering during boot
- Stops the system from running unverified binaries
- Protects against boot kits, rootkits, and other low level malware

### BitLocker

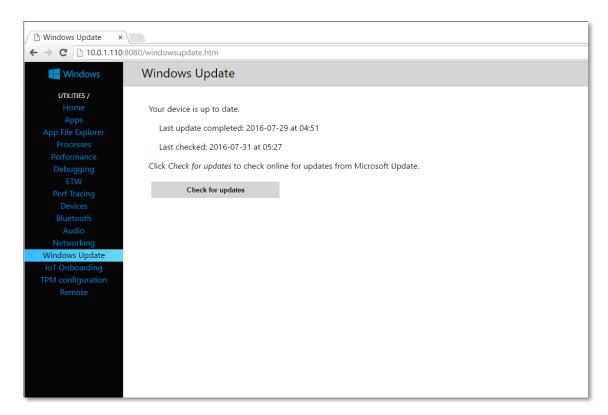
- Lightweight version of BitLocker
- Encryption of user and system files

Internals > Windows Update

Automatic forced update

 Check for updates through "Windows Update" tab of Windows Device Portal

 Pro edition allows deferred updates



## **Attack Surface**

### Attack Surface > Network services

```
Starting Nmap 7.12 (https://nmap.org ) at 2016-07-13 01:33 Malay Peninsula Standard Time

Nmap scan report for 10.0.1.108

Host is up (0.020s latency).

Not shown: 996 closed ports

PORT STATE SERVICE

22/tcp open ssh

135/tcp open msrpc

445/tcp open microsoft-ds
8080/tcp open http-proxy

MAC Address: B8:27:EB:B5:A9:E0 (Raspberry Pi Foundation)

Nmap done: 1 IP address (1 host up) scanned in 3.24 seconds
```

http://<device ip>:8080

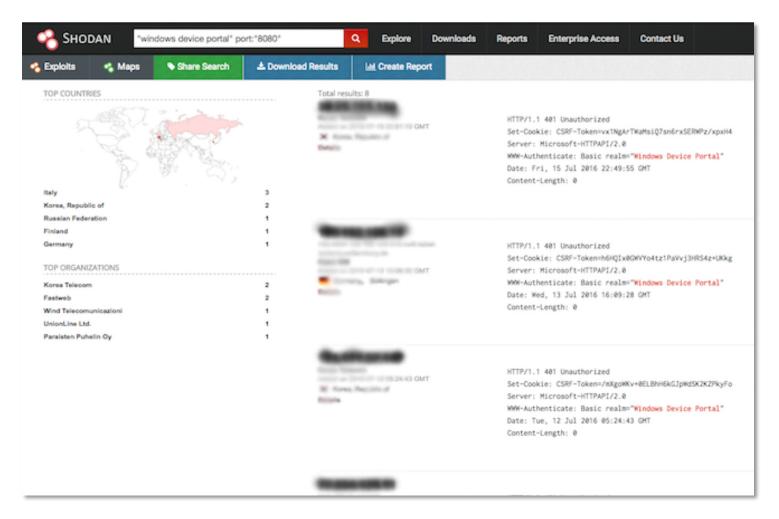
 Files can be found in C:\Windows \WebManagement\www on the device

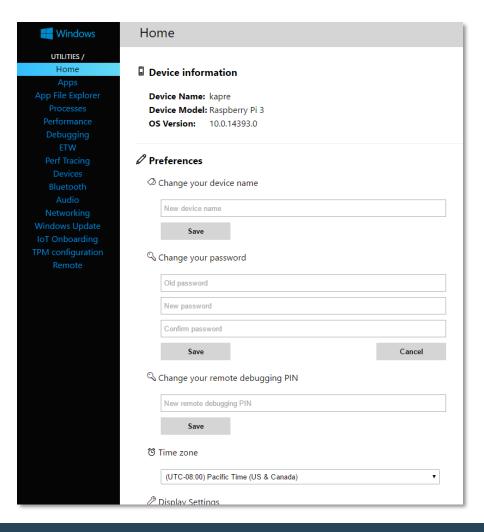
 User name: Administrator, password: p@ssw0rd

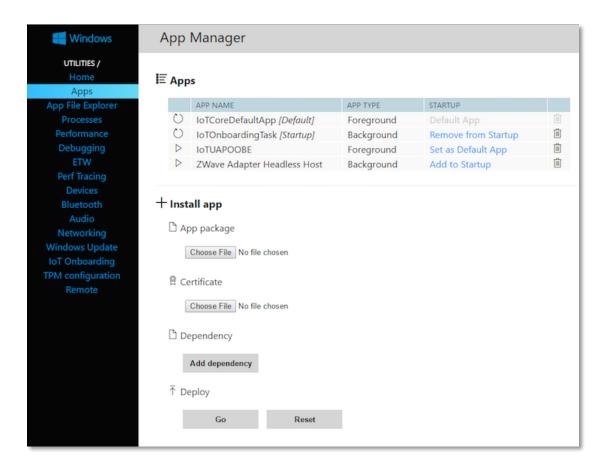
- Built on top of REST APIs
  - <device ip>:8080/ restdocumentation.htm

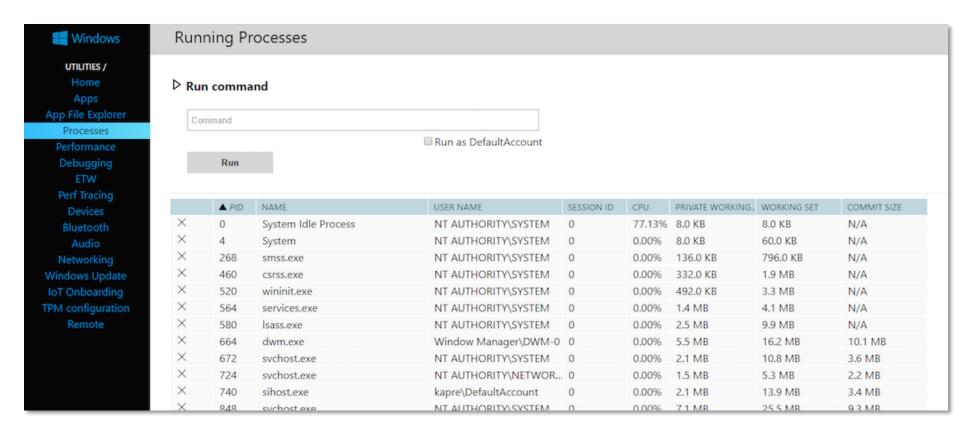
Utility	Function
Home	Device information, change device name/password, timezone settings
Apps	Install/uninstall of apps
App File Explorer	File explorer for installed apps locations
Processes	Running processes list, process memory usage, and process termination
Performance	Real time graphical display of CPU and I/O usage
Debugging	Starting VS remote debugger, downloading of live kernel and process dumps
ETW	Event tracing
Perf Tracing	Trace logging of CPU, disk, and memory usage
Devices	Device manager for peripherals attached to the device
Bluetooth	Bluetooth device search
Audio	Device speaker and microphone volume adjustments
Networking	WiFi configuration
Windows Update	Last update timestamp, check for updates
IoT Onboarding	Internet Connection Sharing settings, SoftAP settings, AllJoyn onboarding settings
TPM Configuration	TPM installation, configuration, and provisioning
Remote	Enable Windows IoT Remote Server

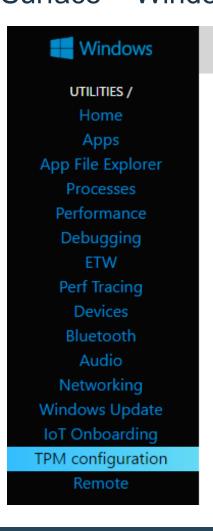
### Attack Surface > Network services











### TPM configuration

### TPM information

There is currently no TPM installed.

Select TPM to install None

Install

### Software TPM Emulator (NoSecurity)

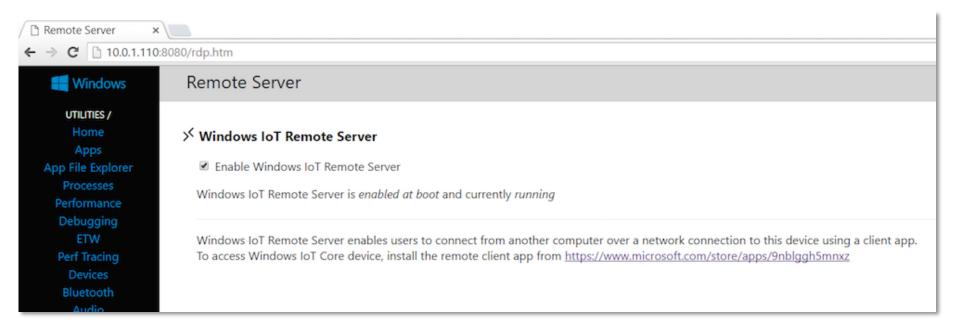
Discrete SPI TIS Class TPM on SPI0CS0
Discrete SPI TIS Class TPM on SPI0CS1
Discrete SPI NationZ TPM on SPI0CS0
Discrete SPI NationZ TPM on SPI0CS1
Discrete I2C TIS Class TPM on I2C1
Discrete I2C NationZ TPM on I2C1

### Attack Surface > Network services

- SSH
  - Enabled by default
  - Starts at boot

- Windows File Sharing
  - Enabled by default
  - Starts at boot

- Windows IoT Remote Server
  - Remote UI client installed from Windows Store
  - Can be enabled through the Remote tab in the Windows Device Portal
  - No authentication
  - NanoRDPServer.exe



### Attack Surface > Device drivers

Drivers for built-in or external peripherals

- Drivers for wireless adapters
  - Wifi
  - Bluetooth
  - ZigBee
  - Z-Wave

Successful exploitation often results in kernel level privilege

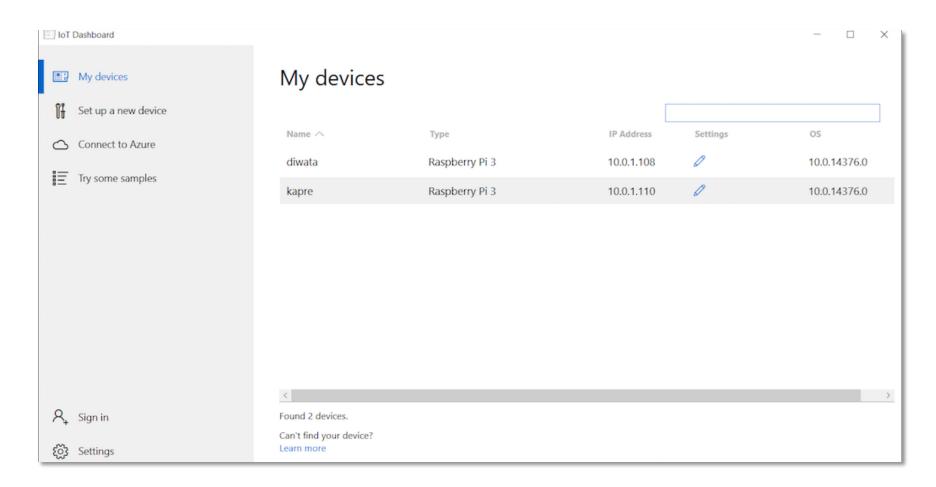
### Attack Surface > Malware

- Password guessing/brute forcing of login credentials
- Vulnerabilities in the network services
- Lateral infection coming from other machines

```
mimikatz 2.1 (x64) built on Jul 11 2016 00:32:57
 .## ^ ##. "A La Vie, A L'Amour"
 ## \ / ## Benjamin DELPY `gentilkiwi` ( benjamin@gentilkiwi.com )
          http://blog.gentilkiwi.com/mimikatz
mimikatz # privilege::debug
Privilege '20' OK
                : DESKTOP-39HUL88
Logon Server : (null)
Logon Time : 7/20/2016 6:15:59 PM
           Username : Administrator
```

# Hacking

### Hacking > Device Discovery



### Hacking > Device Discovery

```
7013 383.345753
                   10.0.1.108
                                     239.0.0.222
                                                               488 6 → 6 Len=446
  7021 388.261044
                   10.0.1.108
                                    239.0.0.222
                                                      UDP
                                                               488 6 → 6 Len=446
                  10.0.1.108
   7029 393.483394
                                     239.0.0.222
                                                      UDP
                                                               488 6 → 6 Len=446
                  10.0.1.108
   7035 398.410021
                                     239.0.0.222
                                                      UDP
                                                               488 6 → 6 Len=446
> Frame 6935: 488 bytes on wire (3904 bits), 488 bytes captured (3904 bits) on interface 0
Ethernet II, Src: Raspberr b5:a9:e0 (b8:27:eb:b5:a9:e0), Dst: IPv4mcast de (01:00:5e:00:00:de)
 Internet Protocol Version 4, Src: 10.0.1.108, Dst: 239.0.0.222
 User Datagram Protocol, Src Port: 6 (6), Dst Port: 6 (6)
 Data (446 bytes)
     01 00 5e 00 00 de b8 27 eb b5 a9 e0 08 00 45 00
                                                 ..^...' .....E.
     01 da 2f c5 00 00 08 11 86 04 0a 00 01 6c ef 00
                                                 ../..... .....1..
     00 de 00 06 00 06 01 c6 10 f0 64 00 69 00 77 00
0020
0030
     61 00 74 00 61 00 00 00 00 00 00 00 00 00 00 00
0040
     0050
     0060
     00 00 00 00 00 00 00 00 00 00 00 00 31 00 30 00
0070
     2e 00 30 00 2e 00 31 00   2e 00 31 00 30 00 38 00
                                                 .0...1. ..1.0.8
0080
     38 00 3a 00 32 00 37 00  3a 00 65 00 62 00 3a 00
                                                 8.:.2.7. :.e.b.:
0090
00a0
     62 00 35 00 3a 00 61 00  39 00 3a 00 65 00 30 00
                                                 b.5.:.a. 9.:.e.0
                                                 :.0.0.:. 0.0....
00b0
     3a 00 30 00 30 00 3a 00
                         30 00 30 00 00 00 00 00
                                                 {.3.c.a. d.0.6.3
00c0
     7b 00 33 00 63 00 61 00  64 00 30 00 36 00 33 00
00d0
     65 00 2d 00 30 00 30 00  30 00 30 00 2d 00 30 00
                                                 e.-.0.0. 0.0.-.0
                                                 0.0.0.-. 0.0.0.0
00e0
     30 00 30 00 30 00 2d 00 30 00 30 00 30 00 30 00
                                                 -.0.0.0. 0.b.9.e
00f0
     2d 00 30 00 30 00 30 00  30 00 62 00 39 00 65 00
     30 00 66 00 63 00 62 00 35 00 7d 00 00 00 00 00
                                                 0.f.c.b. 5.}....
0100
0110
     52 00 61 00 73 00 70 00  62 00 65 00 72 00 72 00
                                                 R.a.s.p. b.e.r.r
0120
     79 00 20 00 50 00 69 00
                         20 00 33 00 00 00 00 00
0130
     0140
     0150
     0160
     0170
     00 00 00 00 31 00 30 00  2e 00 30 00 2e 00 31 00
                                                 ....1.0. ..0...1
     34 00 33 00 37 00 36 00 2e 00 30 00 00 00 00 00
0180
                                                 4.3.7.6. ..0....
```

Offset	Description
0	Device name
0x42	IP address
0x64	MAC address
0x96	Board serial number
0xe6	Device Type
0x14a	OS version
0x1ae	Device architecture

### Hacking > PowerShell

 Remote device administration and configuration

- Built-in and 3<sup>rd</sup> party tools for penetration testing and reversing. Ex:
  - CimSweep
  - Autoruns

```
PS C:\WINDOWS\system32> $CimSessionPi2 = New-CimSession -ComputerName
PS C:\WINDOWS\system32> Get-CSRegistryAutoStart -CimSession $CimSessionPi2
               : HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon
ImagePath
               : IotShell.exe
               : HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon
ImagePath
               : HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon
               : SystemPropertiesPerformance.exe /pagefile
ImagePath
               : HKLM\SYSTEM\CurrentControlSet\Control\Session Manager
ImagePath
               : autocheck autochk *
```

### Hacking > Static analysis

 UWP apps can be found in Data partition (U:\, also linked with C:\Data)

App installed in U:\Programs
 \WindowsApps

Lib DLLs and XBF (binary XAML)

- Assets folder
  - Images
  - Fonts
  - etc

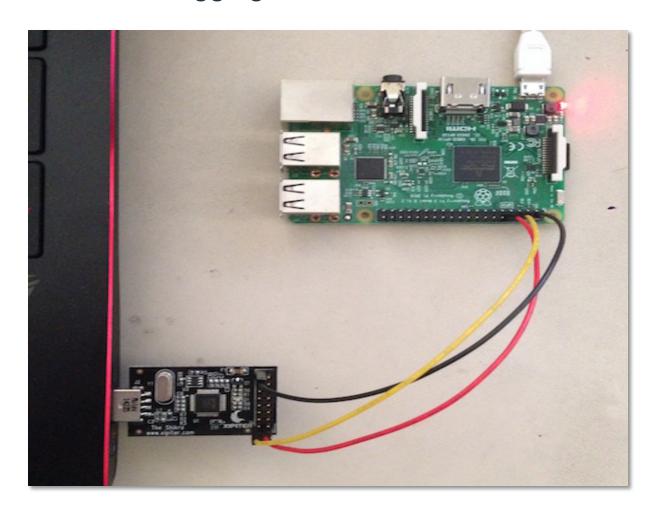
Filename	Description
<app_name>.exe</app_name>	App startup stub
<app_name>.dll</app_name>	App code
AppManifest.xml	UWP app package manifest
AppBlockMap.xml	Cryptographic block hashes for files in package
AppxSignature.p7x	App package digital signature file

### Hacking > Static analysis

```
🌃 🚟
; Section 4. (virtual address 00006000)
; Virtual size
                             : 00000010 (
                                             16.)
 Section size in file : 00000200 (
                                            512.)
 Offset to raw data for section: 00003600
 Flags 60020020: Text Executable Readable
; Alignment : default
; Segment type: Pure code
AREA .text, CODE, ALIGN=4
; ORG 0x406000
CODE16
EXPORT start
start
MOV R12, #RHBinder ShimExeMain
LDR.W PC, [R12]
; End of function start
```

IBM Security

### Hacking > Kernel debugging



### Hacking > Kernel debugging

```
# Enable serial debugging
bcdedit -dbgsettings serial
# Turn on debugging
bcdedit -debug on
```

Get-WMIObject Win32 pnpentity | ? Name -like "\*Serial\*COM\*"

```
: Win32 PnPEntity
  CLASS
                            : CIM LogicalDevice
  SUPERCLASS
                            : CIM ManagedSystemElement
  DYNASTY
 RELPATH
  PROPERTY COUNT
                             : {CIM LogicalDevice, CIM LogicalElement, CIM ManagedSystemElement}
  DERIVATIŌN
                             : DESKTOP-39HUL88
  SERVER
 NAMESPACE
 PATH
                            : \DESKTOP-39HUL88\root\cimv2:Win32 PnPEntity.DeviceID="FTDIBUS\
\(\nabla\) ID 0403+PID 6014+5&3278CBC5&0&3\\0000"
Availability -
                             : USB Serial Port (COM3)
                             : {4d36e978-e325-11ce-bfc1-08002be10318}
ClassGuid
CompatibleID
```

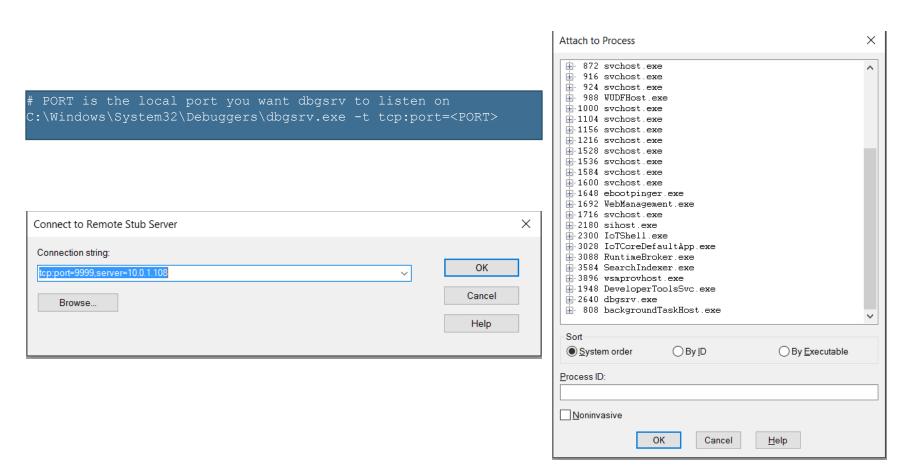
### Hacking > Kernel debugging

```
# PORT is the COM port number used by your USB-to-serial adapter
windbg.exe -k com:port=<PORT>,baud=921600
```

```
Microsoft (R) Windows Debugger Version 10.0.10586.567 X86
Copyright (c) Microsoft Corporation. All rights reserved.
Waiting to reconnect...
```

```
Connected to Windows 10 14393 ARM (NT) Thumb-2 target at (Sun Jul 24 19:32:43.111 2016 (UTC + 8:00)), ptr64 FALSE
Kernel Debugger connection established.
Symbol search path is: srv*
*** ERROR: Symbol file could not be found. Defaulted to export symbols for ntkrnlmp.exe -
Windows 10 Kernel Version 14393 MP (1 procs) Free ARM (NT) Thumb-2
Machine Name:
Kernel base = 0x80c1b000 PsLoadedModuleList = 0x80e07c78
Break instruction exception - code 80000003 (first chance)
       CTRL+BREAK (if you run GUI kernel debugger),
   on your debugger machine's keyboard.
                   THIS IS NOT A BUG OR A SYSTEM CRASH
*** ERROR: Symbol file could not be found. Defaulted to export symbols for ntkrnlmp.exe -
nt!DbgBreakPointWithStatus:
80c40d90 defe
```

### Hacking > User mode debugging



### Hacking > User mode debugging

```
Copyright (c) Microsoft Corporation. All rights reserved.
Executable search path is:
ModLoad: 01110000 011db000
                             C:\windows\system32\WebManagement.exe
ModLoad: 77400000 77565000
                             C:\windows\SYSTEM32\ntdll.dll
                             C:\windows\System32\KERNELBASE.dll
ntdll!DbgBreakPoint:
77422740 defe
PEB at 00928000
    InheritedAddressSpace:
    ReadImageFileExecOptions: No
    BeingDebugged:
    ImageBaseAddress:
                              774eb9e0
    Ldr.Initialized:
    Ldr.InInitializationOrderModuleList: 00c41738 . 00c4fcd0
    Ldr.InLoadOrderModuleList:
                                         00c41810 . 00c4fcc0
    Ldr.InMemoryOrderModuleList:
                                         00c41818 . 00c4fcc8
            Base TimeStamp
         1110000 57898ebe Jul 16 09:32:46 2016 C:\windows\system32\WebManagement.exe
        77400000 57898ba5 Jul 16 09:19:33 2016 C:\windows\SYSTEM32\ntdll.dll
        77270000 57898c4c Jul 16 09:22:20 2016 C:\windows\System32\KERNELBASE.dll
WebManagement+0xa6631:
011b6630 e92d4800 push
                              {r11, lr}
011b6634 46eb
                              r11,sp
011b6636 f000fb65 bl
                              WebManagement+0xa6d04 (011b6d04)
011b663a e8bd4800 pop
                              {r11,lr}
011b663e f7ffbf25 b.w
                              WebManagement+0xa648c (011b648c)
011b6642 0000
011b6644 f24c6c64 mov
                              r12,#0xC664
011b6648 f2c01c1c movt
                              r12, #0x11C
```

### Hacking > Crash Dump

▲ PID

268

□ 0

□ 4

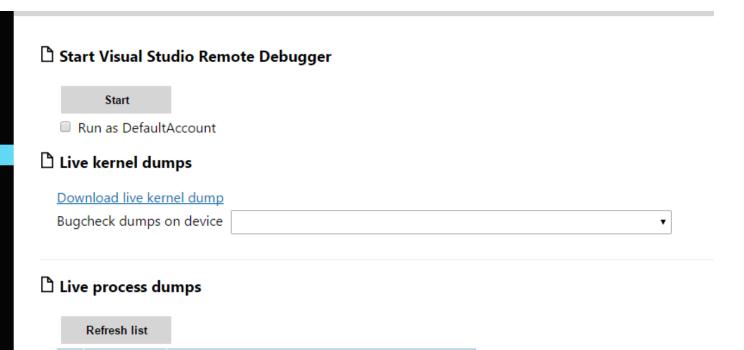
NAME

System

smss.exe

System Idle Process





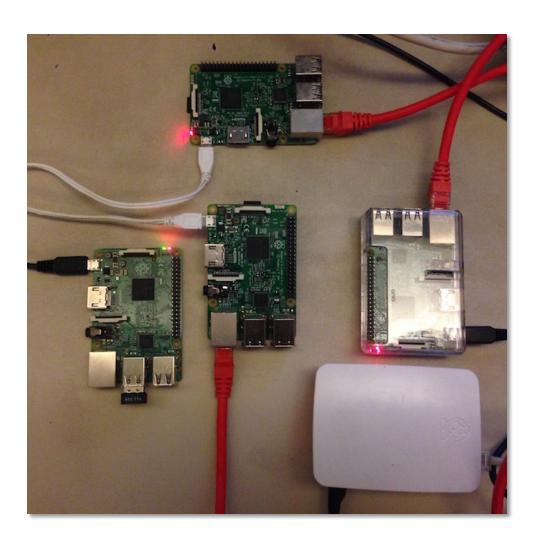
### Hacking > Crash Dump

```
Microsoft (R) Windows Debugger Version 10.0.10586.567 X86
Copyright (c) Microsoft Corporation. All rights reserved.
Loading Dump File [d:\winiot\WebManagement.exe-LiveUM-2016-07-24-12-36-09.dmp]
User Mini Dump File: Only registers, stack and portions of memory are available
Symbol search path is: srv*
Executable search path is:
Windows 10 Version 14376 MP (4 procs) Free ARM (NT) Thumb-2
Product: WinNt, suite: SingleUserTS
Built by: 10.0.14376.0 (rs1 release.160624-1700)
Machine Name:
Debug session time: Mon Jul 25 03:36:09.000 2016 (UTC + 8:00)
Systém Uptime: not available
Process Uptime: 1 days 4:48:37.000
Loading unloaded module list
Cannot read PEB32 from WOW64 TEB32 fffffffff - Win32 error 0n30 Unable to load image C:\Windows\System32\ntdl1.dl1, Win32 error 0n2
*** WARNING: Unable to verify timestamp for ntdll.dll
ntdll!NtWaitForSingleObject+0x6:
*** WARNING: Unable to verify timestamp for KERNELBASE.dll
77320ab6 4770
                    id: 698 examine name: C:\Windows\System32\WebManagement.exe
PEB at 032f8000
     InheritedAddressSpace:
    ReadImageFileExecOptions: No
     BeingDebugged:
                                   00a00000
     ImageBaseAddress:
                                   773eb9e0
    Ldr.Initialized:
    Ldr.InInitializationOrderModuleList: 034a1730 . 034ae758
    Ldr.InLoadOrderModuleList: 034a1808 . 034ae748
Ldr.InMemoryOrderModuleList: 034a1810 . 034ae750
              Basē TimeStamp
         a00000 576dee48 Jun 25 10:36:56 2016 C:\windows\system32\WebManagement.exe 77300000 576deb18 Jun 25 10:23:20 2016 C:\windows\SYSTEM32\ntdll.dll
         76f20000 576debe7 Jun 25 10:26:47 2016 C:\windows\System32\KERNELBASE.dll
         770b0000 576debda Jun 25 10:26:34 2016 C:\windows\System32\combase.dll 76ce0000 576deb16 Jun 25 10:23:18 2016 C:\windows\System32\ucrtbase.dll
         76e30000 576ded32 Jun 25 10:32:18 2016 C:\windows\System32\RPCRT4.dll
76de0000 576dee1b Jun 25 10:36:11 2016 C:\windows\System32\kernel32legacy.dll
         76d90000 576deeaa Jun 25 10:38:34 2016 C:\windows\System32\bcryptPrimitives.dll
```

### Hacking > Fuzzing

- Current Approach
  - Old school
  - REST APIs to control device

- Future Approach
  - Corpus driven fuzzing
  - WinAFL



## Recommendations

### Recommendations

### Segment your network

- · Mitigates lateral infection
- · Incident isolation and cleanup

### Protect network services

- Use built-in firewall
- Disable unnecessary services

## Change default Administrator password

• Eliminates most malware infection attempts today

## Use devices supporting TPM

- Minnowboard +Dragonboard
- Raspberry Pi + Discrete TPM

## Take advantage of available security features

- Enable Secure Boot
- Enable BitLocker

## Conclusion

### Conclusion

Windows 10 IoT Core's features makes it an attractive alternative to today's IoT OS

Attack surface is smaller than other computing devices, but if IoT services are factored in, will be bigger

Vendors/makers should be careful about misconfigurations

More security research needed/encouraged

## Questions?



# THANK YOU

### **FOLLOW US ON:**



ibm.com/security



securityintelligence.com



xforce.ibmcloud.com



@ibmsecurity



youtube/user/ibmsecuritysolutions

© Copyright IBM Corporation 2016. All rights reserved. The information contained in these materials is provided for informational purposes only, and is provided AS IS without warranty of any kind, express or implied. IBM shall not be responsible for any damages arising out of the use of, or otherwise related to, these materials. Nothing contained in these materials is intended to, nor shall have the effect of, creating any warranties or representations from IBM or its suppliers or licensors, or altering the terms and conditions of the applicable license agreement governing the use of IBM software. References in these materials to IBM products, programs, or services do not imply that they will be available in all countries in which IBM operates. Product release dates and / or capabilities referenced in these materials may change at any time at IBM's sole discretion based on market opportunities or other factors, and are not intended to be a commitment to future product or feature availability in any way. IBM, the IBM logo, and other IBM products and services are trademarks of the International Business Machines Corporation, in the United States, other countries or both. Other company, product, or service names may be trademarks or service marks of others.

Statement of Good Security Practices: IT system security involves protecting systems and information through prevention, detection and response to improper access from within and outside your enterprise. Improper access can result in information being altered, destroyed, misappropriated or misused or can result in damage to or misuse of your systems, including for use in attacks on others. No IT system or product should be considered completely secure and no single product, service or security measure can be completely effective in preventing improper use or access. IBM systems, products and services are designed to be part of a lawful, comprehensive security approach, which will necessarily involve additional operational procedures, and may require other systems, products or services to be most effective.

IBM DOES NOT WARRANT THAT ANYSYSTEMS, PRODUCTS OR SERVICES ARE IMMUNE FROM, OR WILL MAKE YOUR ENTERPRISE IMMUNE FROM, THE MALICIOUS OR ILLEGAL CONDUCT OF ANY PARTY.

