



Compass Security

[The ICT-Security Experts]



Windows Phone

Windows Phone 8.1

[Beer Talk – Berlin – 2015/07/21]

Stephan Sekula

Compass Security
Deutschland GmbH
Tauentzienstr. 18
De-10789 Berlin

Tel. +49 30 21 00 253-0
Fax +49 30 21 00 253-69
team@csnc.de
www.csnc.de

May I introduce myself?



Stephan Sekula

- ✦ With Compass since August 2013
- ✦ Career: Studied Computer Science, worked in research, now gathering practical experience
- ✦ Expertise
 - ✦ War Dialing
 - ✦ Social Engineering basics (from Psychology minor)

Hobbies

- ✦ Cooking & baking
- ✦ Bicycling
- ✦ Climbing
- ✦ IT-Security



Agenda



Introduction

Windows Aspects

- ✦ Windows Environment
- ✦ Attack Surface
- ✦ Breaking Out

Mobile Aspects

- ✦ Sandboxing & Encryption

Findings

- ✦ MDM Integration
- ✦ Wi-Fi Sense
- ✦ Low Level Storage API

Conclusion

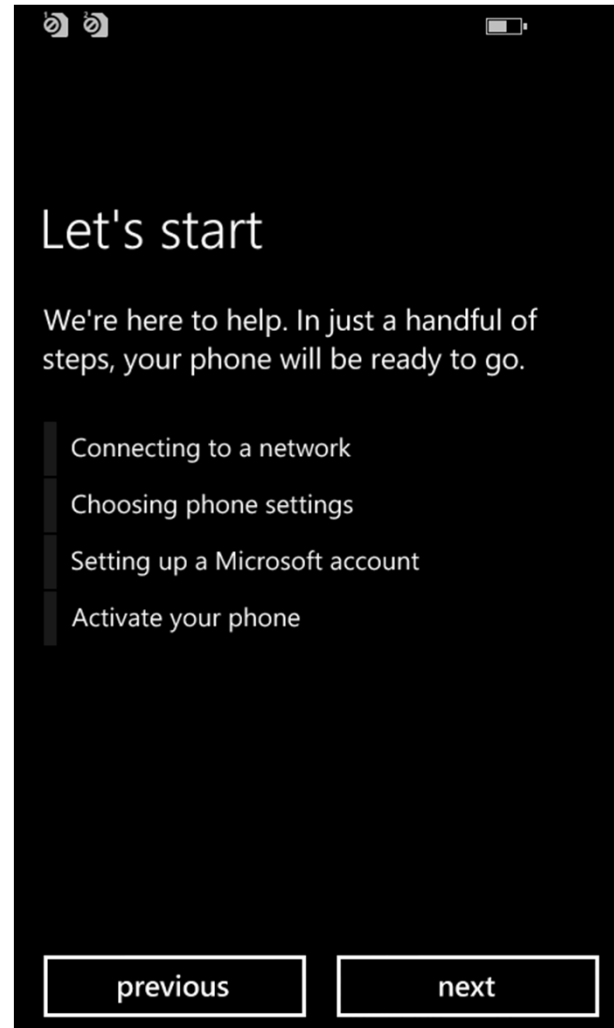
Microsoft

- ✦ Major player in home computing, servers, software, and entertainment
- ✦ Still very new to the mobile sector
- ✦ But attempting to catch up with the acquisition of Nokia
- ✦ Still understands / answers companies' needs best

Main focus of analysis

- ✦ Windows Phone platform itself

Let's get started



Agenda



Introduction

Windows Aspects

- ✦ Windows Environment
- ✦ Attack Surface
- ✦ Breaking Out

Mobile Aspects

- ✦ Sandboxing & Encryption

Findings

- ✦ MDM Integration
- ✦ Wi-Fi Sense
- ✦ Low Level Storage API

Conclusion

Crash dumps are always useful and a good start...

```
ALLUSERSPROFILE=C:\Data\ProgramData
APPDATA=C:\Data\Users\DefApps\AppData\Roaming
CommonProgramFiles=C:\Program Files\Common Files
COMPUTERNAME=Windows Phone
ComSpec=C:\windows\system32\cmd.exe
FP_NO_HOST_CHECK=NO
LOCALAPPDATA=C:\Data\Users\DefApps\AppData\Local\Packages\8dd8d60d-8b28-4e52-b113-
c2aac34b9ac3_yhxz8gp8y0q0t\AC
NUMBER_OF_PROCESSORS=4
OS=Windows_NT
Path=C:\windows\system32;C:\windows;C:\Programs\CommonFiles\System;C:\lwt;C:\data\test\bin;
PATHEXT=.COM;.EXE;.BAT;.CMD;.VBS;.VBE;.JS;.JSE;.WSF;.WSH;.MSC
PROCESSOR_ARCHITECTURE=ARM
...
ProgramData=C:\Data\ProgramData
ProgramFiles=C:\Program Files
PUBLIC=C:\Data\Users\Public
SystemDrive=C:
SystemRoot=C:\windows
TEMP=C:\Data\Users\DefApps\AppData\Local\Packages\8dd8d60d-8b28-4e52-b113-
c2aac34b9ac3_yhxz8gp8y0q0t\AC\Temp
TMP=C:\Data\Users\DefApps\AppData\Local\Packages\8dd8d60d-8b28-4e52-b113-
c2aac34b9ac3_yhxz8gp8y0q0t\AC\Temp
USERDOMAIN=Windows Phone
USERNAME=DefApps
USERPROFILE=C:\Data\Users\DefApps
windir=C:\windows
```

(Ab)Use of Windows Utilities and Features

- ✦ Is it possible to gather information or perform undesired actions using built-in features?

Application Attack Surface

- ✦ Can apps such as Internet Explorer be used to run unwanted code?

Development and APIs

- ✦ Can APIs be leveraged to execute malicious code?

A Windows desktop is user (and attacker) friendly

- ✦ Lots of information (event logs, detailed error messages, ...)
- ✦ Lots of settings to influence (control panel, file & registry access, ...)
- ✦ Built-in programs and features (notepad, sticky keys for accessibility, ...)
- ✦ Various ways to execute code (bat, vbs, WMI, PowerShell, compilers, ...)

Windows Phone exposes

- ✦ Very little information or settings are available
- ✦ No interesting default apps
- ✦ No possibility to "run" stuff
- ✦ No sticky keys
- ✦ It is impossible to e.g. get the UEFI settings details of the phone...

Internet Explorer is the most interesting app on the phone

User influence / interaction

- ✦ High privileges on the phone
- ✦ Increased attack surface

Failed abuse scenarios

- ✦ Run VBScript within the browser
- ✦ Browse the local file system using file:///
- ✦ SMB connect-back from the phone to the attacker
- ✦ No way to download and execute e.g. .bat, .exe, or .vbs files
- ✦ Link files (.lnk) are not executed, either



We can't download this file, because
Windows Phone doesn't support this file
type.

Can't complete

Can't open file Ink_DriveC.lnk.
Error code: -2147024809. You can mention this
code when providing feedback.

ok

If no app provides the desired feature, develop your own!

The C++ and .NET APIs are trimmed down & restricted, preventing breaking out / unwanted actions

Failed abuse scenarios

- ✦ Controlling processes or threats to fork new content within an application
- ✦ Running arbitrary commands using Shell.Execute
- ✦ Accessing WMI (Windows Management Instrumentation) to gather information and execute arbitrary commands
- ✦ Running PowerShell for the same reasons

Of course, not all options have been explored so far, e.g.

- ✦ Is arbitrary execution of commands possible, via e.g. Lambda expressions?
- ✦ Can the restricted APIs be abused?
(e.g. attempt to load an assembly not present within the Windows Phone SDK)
- ✦ In-depth audit of the Protected Data / Vault feature
- ✦ Study of the AppContainer and SIDs separation
- ✦ Understand the steps involved in the application signing process
(and their capabilities restrictions)
- ✦ Subversion of accorded capabilities
(capabilities seem to be labels assigned to a given process).
- ✦ Content of C:\WTT
- ✦ Corruption via the video driver e.g. within the browser (WebGL)
- ✦ ...

So from a Windows perspective, there is not much attack surface

Some information leaks from application crash dumps, e.g.

- ★ User executing the app / App Container context
- ★ List of defined drives:
 - ★ C:\
 - ★ D:\ (probably SD card)
 - ★ U:\ (probably a mapping to C:\data\.
 - ★ PATH variable contains unknown folder C:\WTT\.
 - ★ Data seems shared via C:\Data\Share
 - ★ C:\windows\system32\cmd.exe does not exist

Agenda



Introduction

Windows Aspects

- ✦ Windows Environment
- ✦ Attack Surface
- ✦ Breaking Out

Mobile Aspects

- ✦ Sandboxing & Encryption

Findings

- ✦ MDM Integration
- ✦ Wi-Fi Sense
- ✦ Low Level Storage API

Conclusion

Sandboxing

- ✦ Attack surface reduction (Least Privilege Principle)
- ✦ User consent and control (Capabilities)
- ✦ Isolation (AppContainer, dedicated SIDs)

Malware Resistance

- ✦ UEFI, Trusted / Secure Boot
- ✦ System and app integrity (code signing)
- ✦ Windows Phone Store (automated malware scan)

Exploit Mitigation

- ✦ Address Space Layout Randomization (ASLR)
- ✦ Data Execution Prevention (DEP)

Encryption

- ✦ BitLocker (AES-128, TPM)

AppContainer

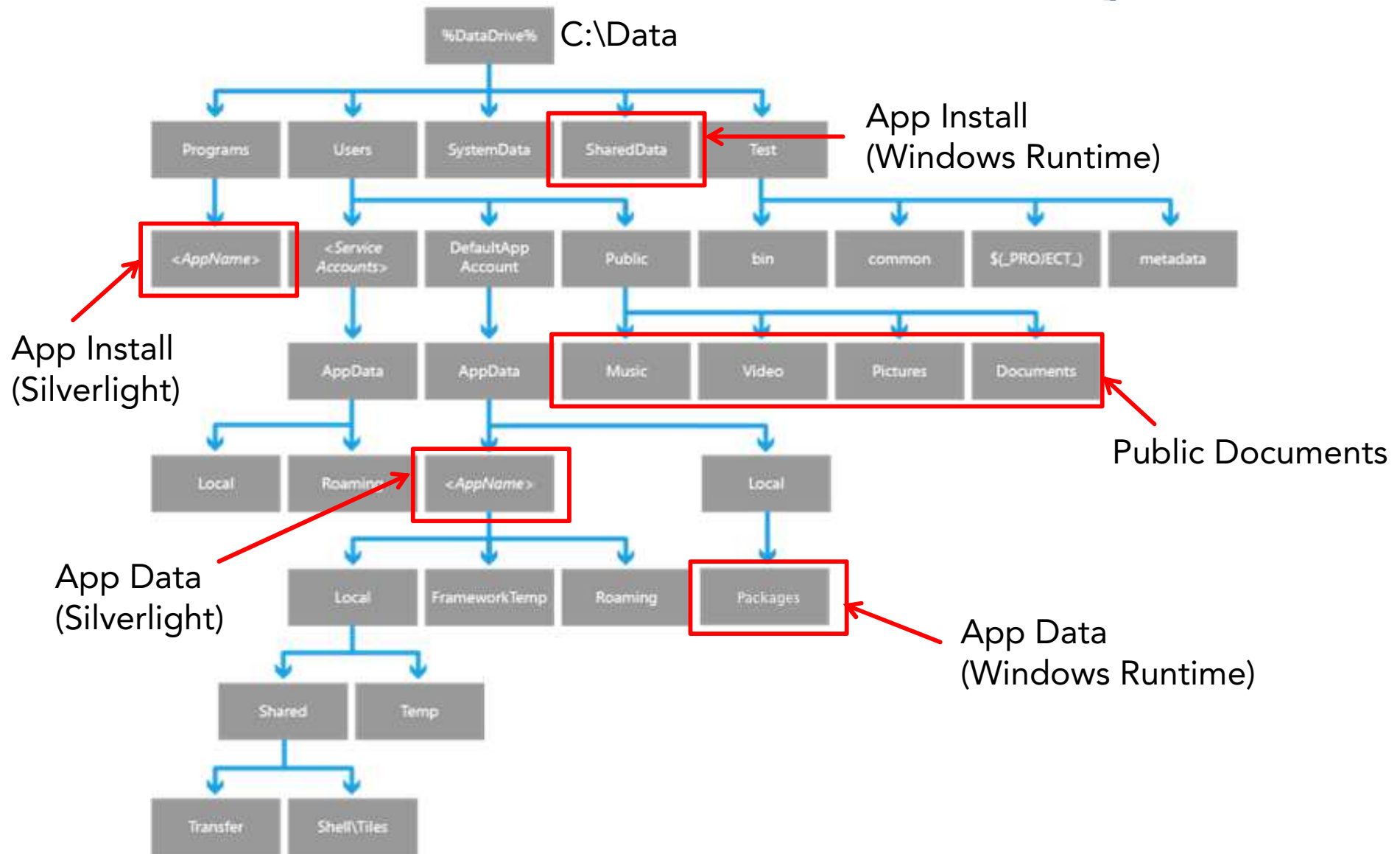
- ✦ Isolation
- ✦ Credentials
- ✦ Roaming
- ✦ Data access
- ✦ Sharing data
- ✦ Encrypting data

Capabilities

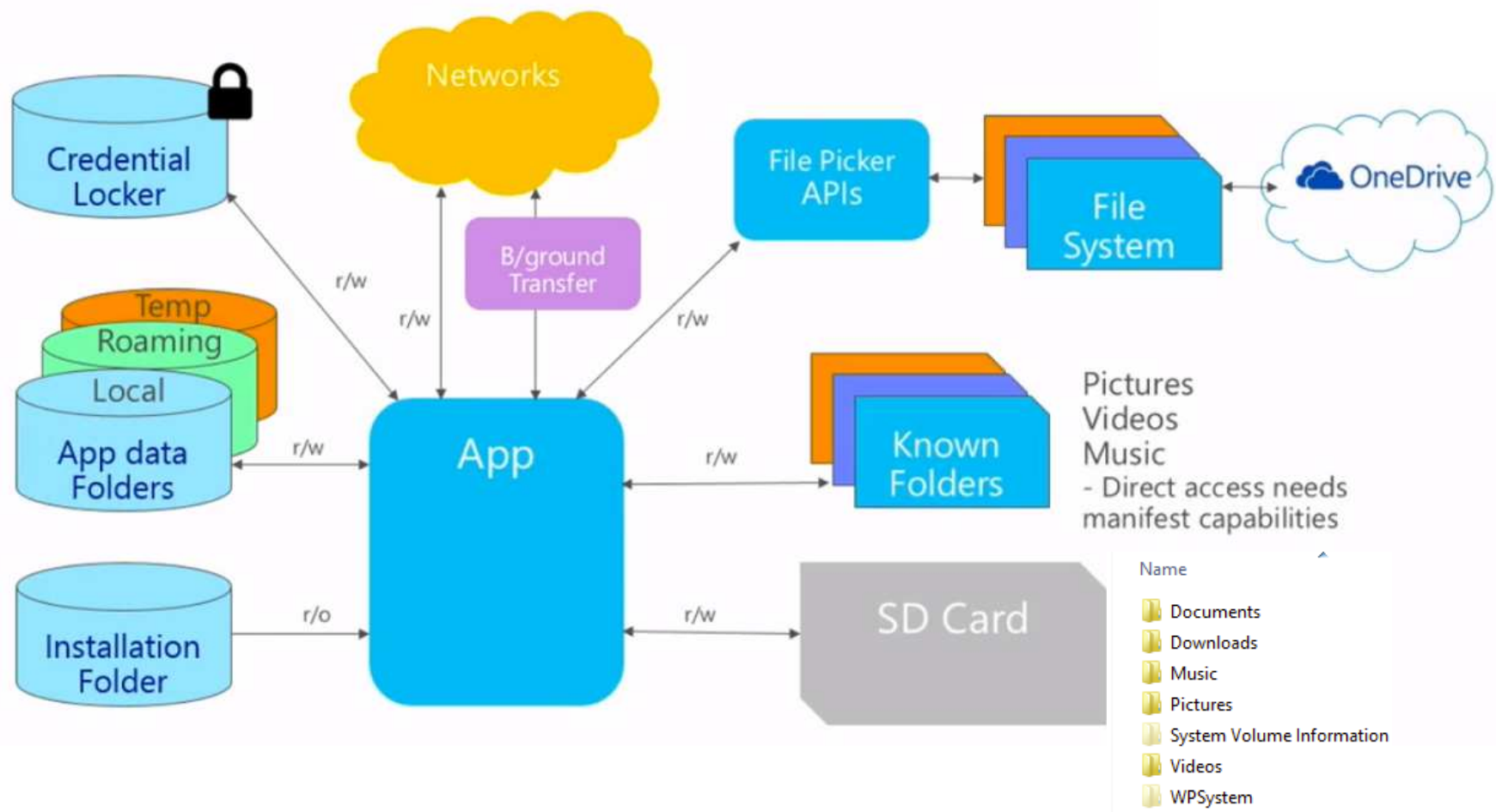
Restricted APIs

- ✦ Isolated storage

File System Overview



Locations where apps can access data



<http://channel9.msdn.com/Series/Building-Apps-for-Windows-Phone-8-1/09>

Storing Credentials

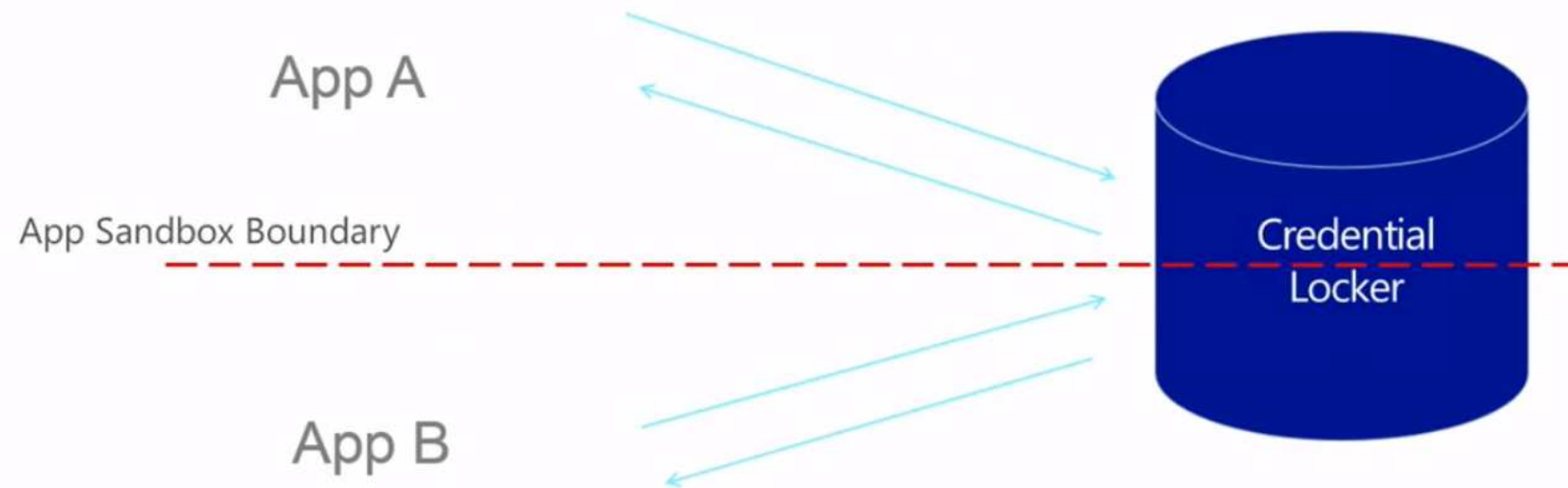
Secure storage & roaming of credentials



Isolation

Apps can only access their own credentials

username / password
pairs only



Example:

```
var vault = new PasswordVault();  
PasswordCredential cred = new PasswordCredential("account", username, password);  
vault.Add(cred);
```

Roaming

Sharing data e.g. credentials across devices



Roaming

Credentials roam across trusted devices



Sharing data between apps works using:

- URI association, where the registered app obtains the data stored in the URI



- File association, where the registered app obtains the file content



- Share contract, allowing custom DataPackages to be shared



Disk Encryption using BitLocker is **disabled** by default

- ✦ End-user cannot enable or disable encryption
- ✦ Can only be activated through ActiveSync or MDM policy

Applications can use DPAPI to protect confidential data

- ✦ DPAPI (Data Protection API) generates and stores a cryptographic key by using the user and device credentials
- ✦ Every app gets its own encryption key, which is created upon the app's first launch
- ✦ Keys are persistent across app updates

Software capabilities

- ✦ Capability elements are entries in the app manifest file
- ✦ User is notified upon installation of an app which capabilities are required
- ✦ E.g. location services

Hardware requirements

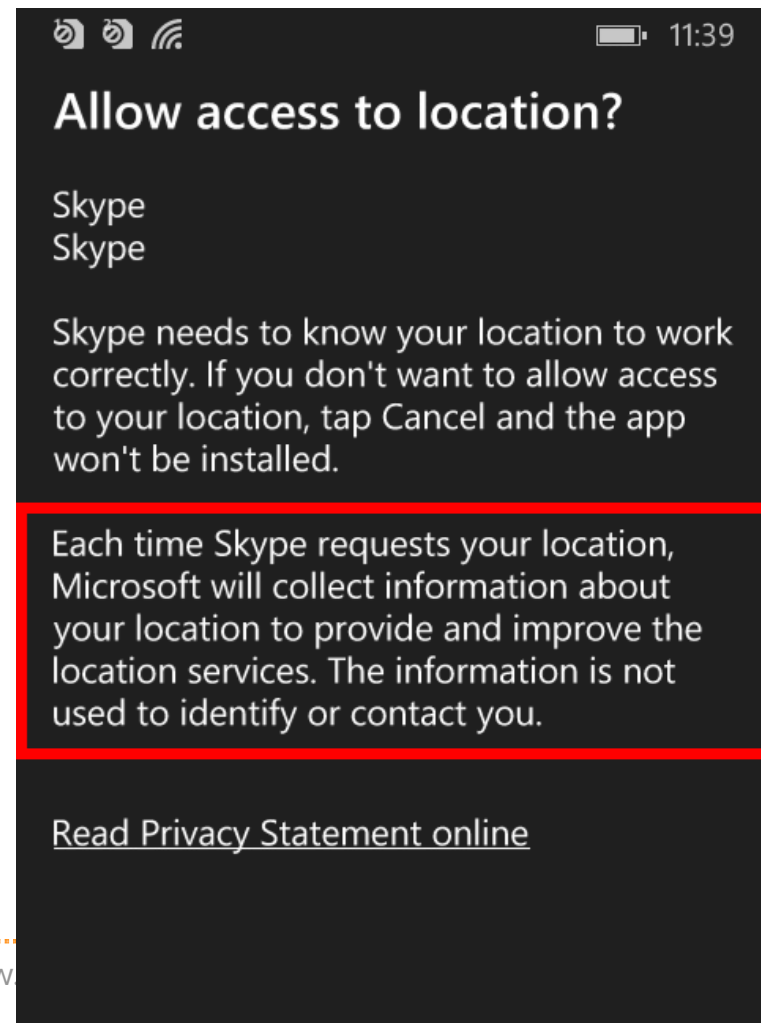
- ✦ Optional entry in the app manifest file
- ✦ Specifies hardware required for running the app
- ✦ E.g. Near Field Communication (NFC)

Functional capabilities

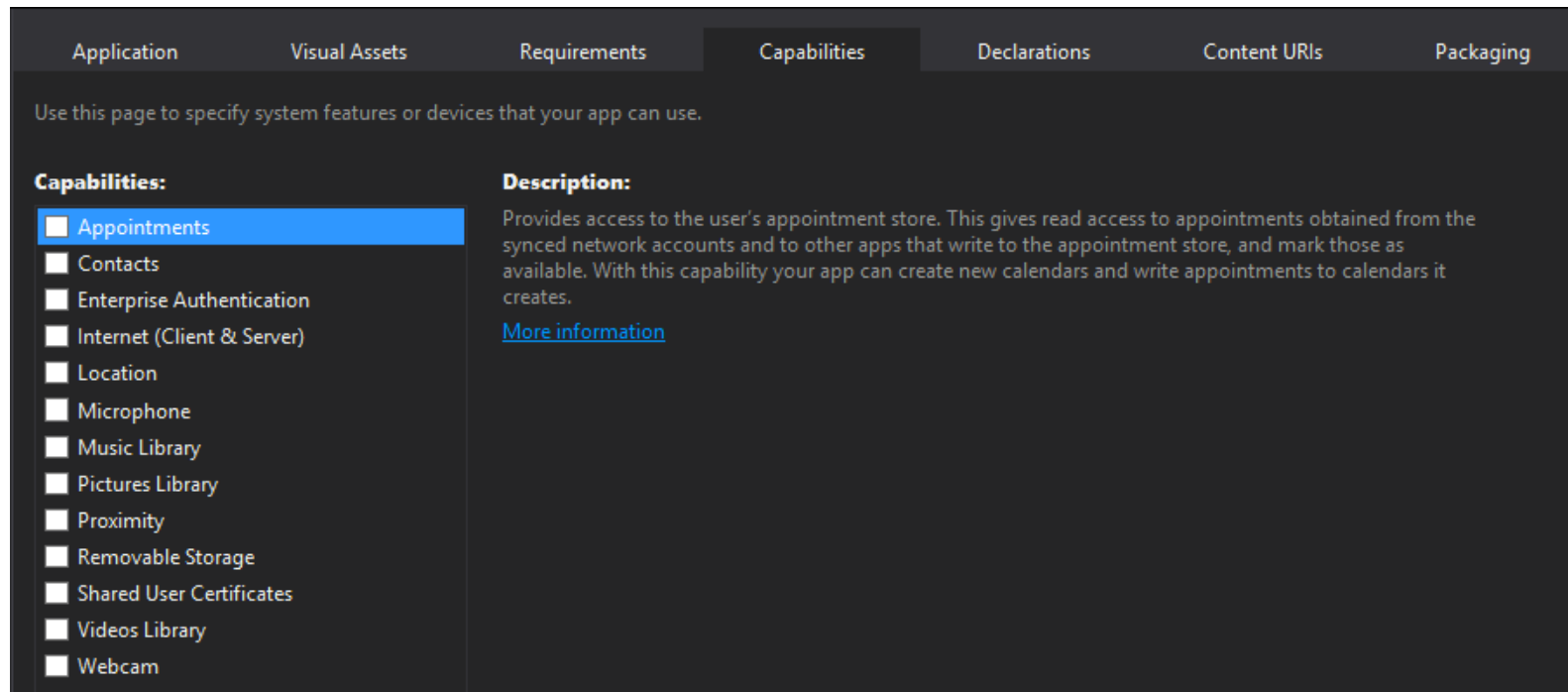
- ✦ Optional entry in the app manifest file
- ✦ Indicates that an app requires a hardware capability of the phone which is present, but not automatically granted
- ✦ E.g. higher memory limits (in Windows Phone 8.0 only)

Capabilities listed in the app manifest

- ✦ Displayed to the user upon installation
- ✦ Some capabilities are prominently displayed (e.g., location services)



Setting capabilities using Microsoft Visual Studio 2013 Express:



Note: When testing apps using the Windows Phone emulator the capabilities are granted automatically, even when not included in the app manifest

Capabilities Overview



Appointments

Allows an app to access the calendar and appointment info.

Camera

Allows an app to access the built-in camera.

Compass

Allows an app to access the built-in compass, if available.

Contacts

Allows an app to access the contact info.

Data services

Your phone's cellular data or Wi-Fi connection.

Gyroscope

Allows an app to access the built-in gyroscope, if available.

Location services

The approximate location.

Libraries

Allows an app to access all photos, music, and videos on your phone.

Microphone

Allows an app to record audio and to use Speech features.

Movement sensor

Allows an app to access the motion sensor.

Proximity

Allows access to the Bluetooth, Wi-Fi, and near field communication (NFC) capabilities.

Owner identity

An anonymous identifier that allows an app to distinguish one person from another, but provides no personal info.

Phone identity

A unique device identifier that allows an app to distinguish one phone from another.

Push notification service

Notifications that an app automatically sends to your phone.

Ringtones

Allows an app to access the ringtones.

SD card

Allows an app access to the SD card.

Speech recognition

Allows an app to access Speech features.

Wallet

Allows an app to access items in your Wallet or to make payments.

Web browser

Allows an app to access the web browser.

Xbox

Allows an app to access the Xbox service or your account info.

Capabilities – WhatsApp



social

WhatsApp



Free

★★★★★
29190 reviews

By installing you agree to the [Terms of Use](#) and other terms

App requires

appointments
contacts
phone identity
owner identity
video and still capture
location services
maps
music library
photos library
media playback
microphone
data services
phone dialer
push notification service
movement and directional sensor
VOIP calling
web browser component
HD720P (720x1280)
WVGA (480x800)
WXGA (768x1280)
appointments
Proximity
SD card
internet connection
videos library
photo, music, and video libraries
camera

Locations all apps can access:

- ✦ Application install directory (read only)
- ✦ Application data locations (local, roaming and temporary directories are created upon app installation)
- ✦ Removable devices (SD card; access is limited to specific file types)
- ✦ User's Downloads folder

Locations requiring additional capabilities in the app manifest:

- ✦ Libraries (Documents, Music, Pictures, Videos)
- ✦ Removable devices (SD card)
- ✦ Homegroup libraries
- ✦ Media server devices (DLNA)
- ✦ Universal Naming Convention (UNC) folders

Agenda



Introduction

Windows Aspects

- ✦ Windows Environment
- ✦ Attack Surface
- ✦ Breaking Out

Mobile Aspects

- ✦ Sandboxing & Encryption

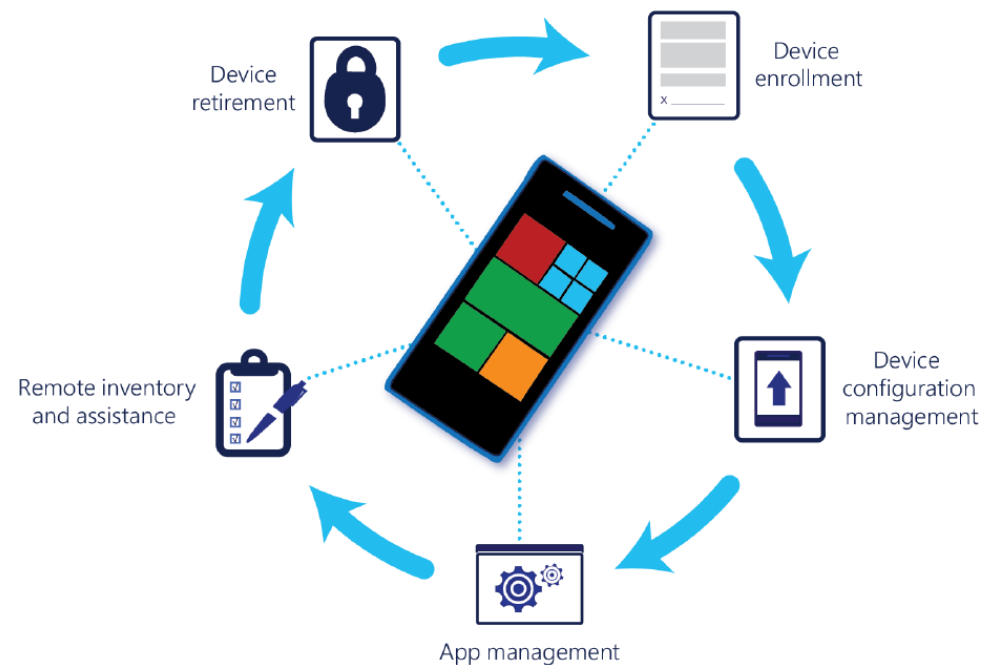
Findings

- ✦ MDM Integration
- ✦ Wi-Fi Sense
- ✦ Low Level Storage API

Conclusion

Push communication

- ✦ To distribute policies and configuration standards
- ✦ Periodically contacts the MDM server to:
 - ✦ Download configurations, apps, updates
 - ✦ Upload asset information



Device Configuration Management

- ✦ Configuration policies
- ✦ Access management
- ✦ Storage management
- ✦ Wi-Fi network / VPN / certificate management
- ✦ Email account / message management

App Management

- ✦ Windows Phone Store apps
- ✦ Side loaded apps
- ✦ Allow / deny apps

Remote Inventory

- ✦ Remote inventory / assistance (lock / PIN reset)

Device Retirement

MDM Integration – Policies



Password policy

Policies that MDM and EAS support	Policies that only MDM supports
Simple password	Disable cellular data roaming
Alphanumeric password	Disable Location
Minimum password length	Disable NFC
Minimum password complex characters	Disable Microsoft Account
Password expiration	Disable roaming between Windows devices
Password history	Disable custom email accounts
Device wipe threshold	Disable screen capture
Inactivity timeout	Disable copy & paste functionality
Device encryption	Disable share and save as
Disable removable storage card	App Allow/Deny list
Disable Camera	Disable Microsoft Store
Disable Bluetooth	Disable development unlock (side loading)
Disable Wi-Fi	Disable Internet Explorer
Disable Sync via USB	Disable Internet Sharing over Wi-Fi
	Disable Wi-Fi Off loading
	Disable Manual Configuration of Wi-Fi Profiles
	Disable Wi-Fi Hotspot reporting
	Disable VPN when Roaming over Cellular
	Disable VPN over Cellular
	Disable mdm un-enrollment and soft factory reset
	Disable Wi-Fi credential sharing
	Lock screen notification controls
	Disable telemetry data submission

Automatically establishes Wi-Fi connections

- ✦ Based on crowdsourcing (Wi-Fi networks other Windows Phone users have connected to)
- ✦ Automatically accepts the Wi-Fi's Terms of Use
- ✦ Automatically provides additional information required to connect (email address, phone number, etc.)
- ✦ Automatically shares Wi-Fi credentials with:
 - ✦ Facebook friends
 - ✦ Outlook.com contacts
 - ✦ Skype contacts

How to prevent users from sharing credentials:

- ✦ Add `_optout` to the Wi-Fi's SSID
- ✦ Problem: Google's `_nomap` suffix

<https://www.windowsphone.com/en-us/how-to/wp8/connectivity/wi-fi-sense-faq>

▲ Win32 storage APIs supported on Windows Phone 8

Windows Phone 8 supports the following Win32 storage APIs for working with the local folder. For the full list of supported Win32 APIs, see [Supported Win32 APIs for Windows Phone 8](#).

- CopyFile2
- CreateDirectoryW
- CreateFile2
- DeleteFileW
- FindClose
- FindFirstFileExW
- FindNextFileW
- FlushFileBuffers

Local folder only?



Device has to be registered / developer unlocked to deploy apps locally (side loading)

Our test app can now access files / pipes etc. outside of the “official” folders

The app can also access documents stored by another app when knowing the path

Does not work anymore if app is signed and distributed via Windows Phone Store

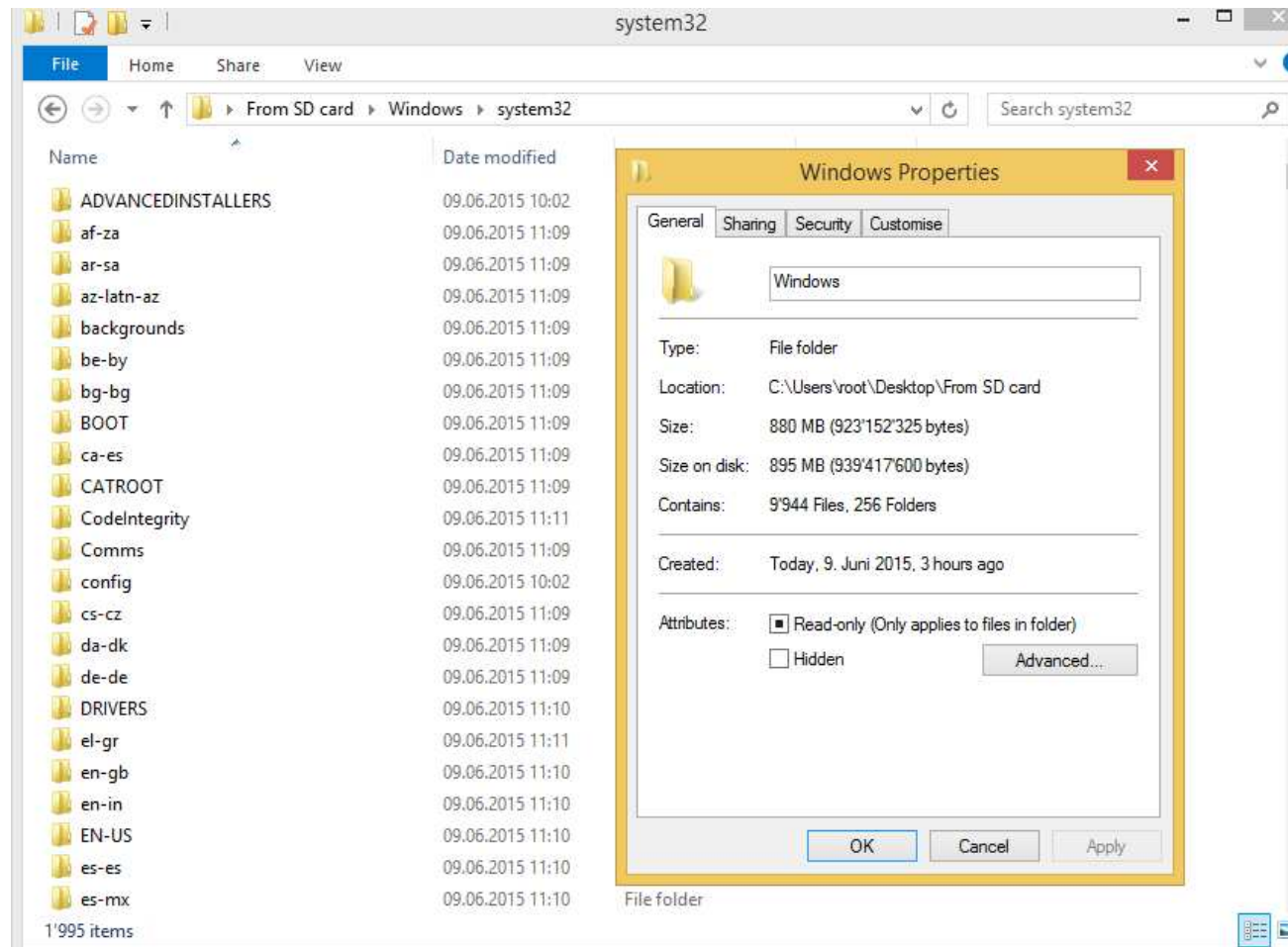
We managed to brick our test phone and had to perform a full reset...

Open Research

Analysis of extracted information



Extracted ~10,000 operating system files

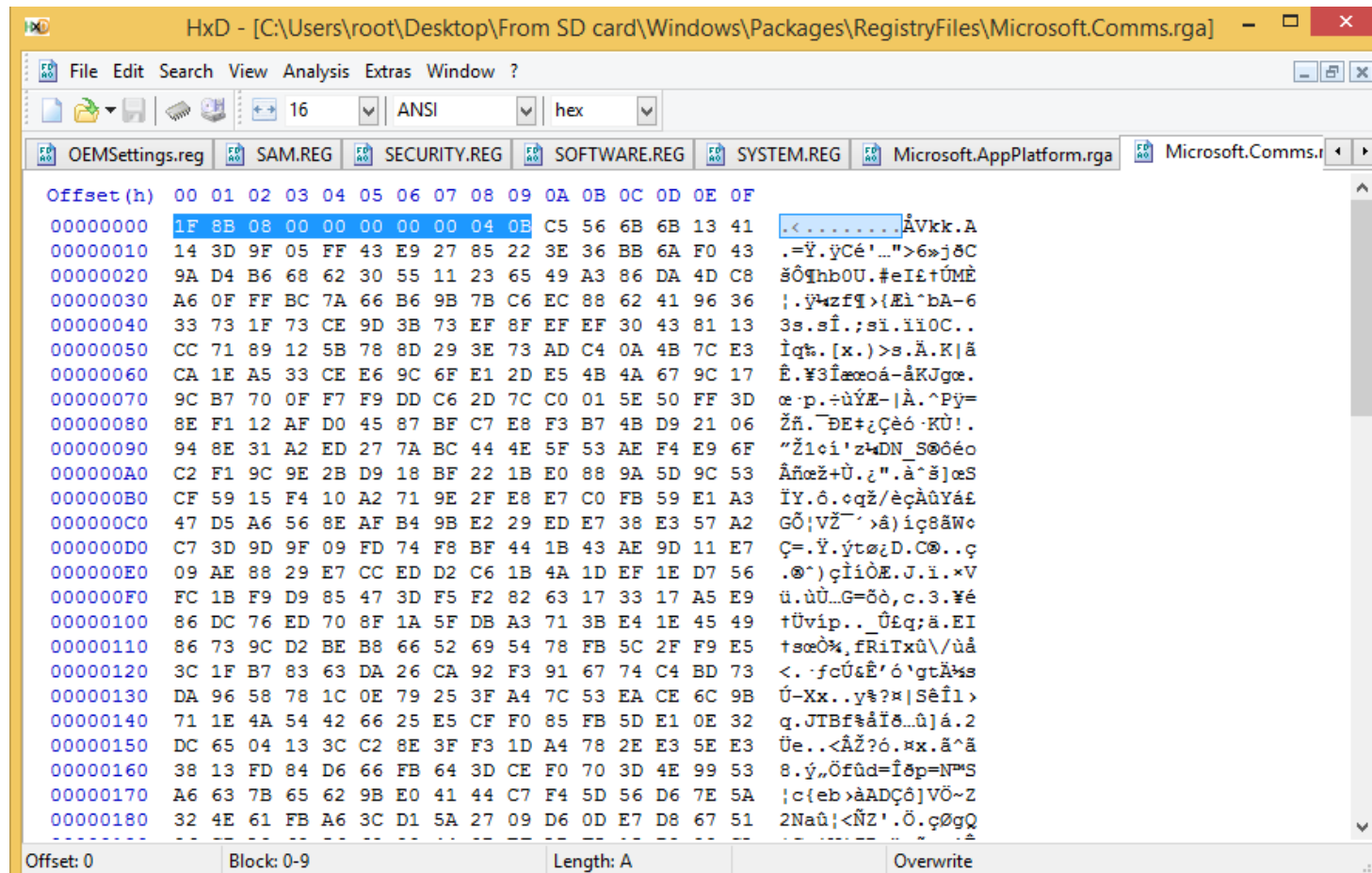


Open Research

Analysis of extracted information



Some documents seem to be encrypted ...

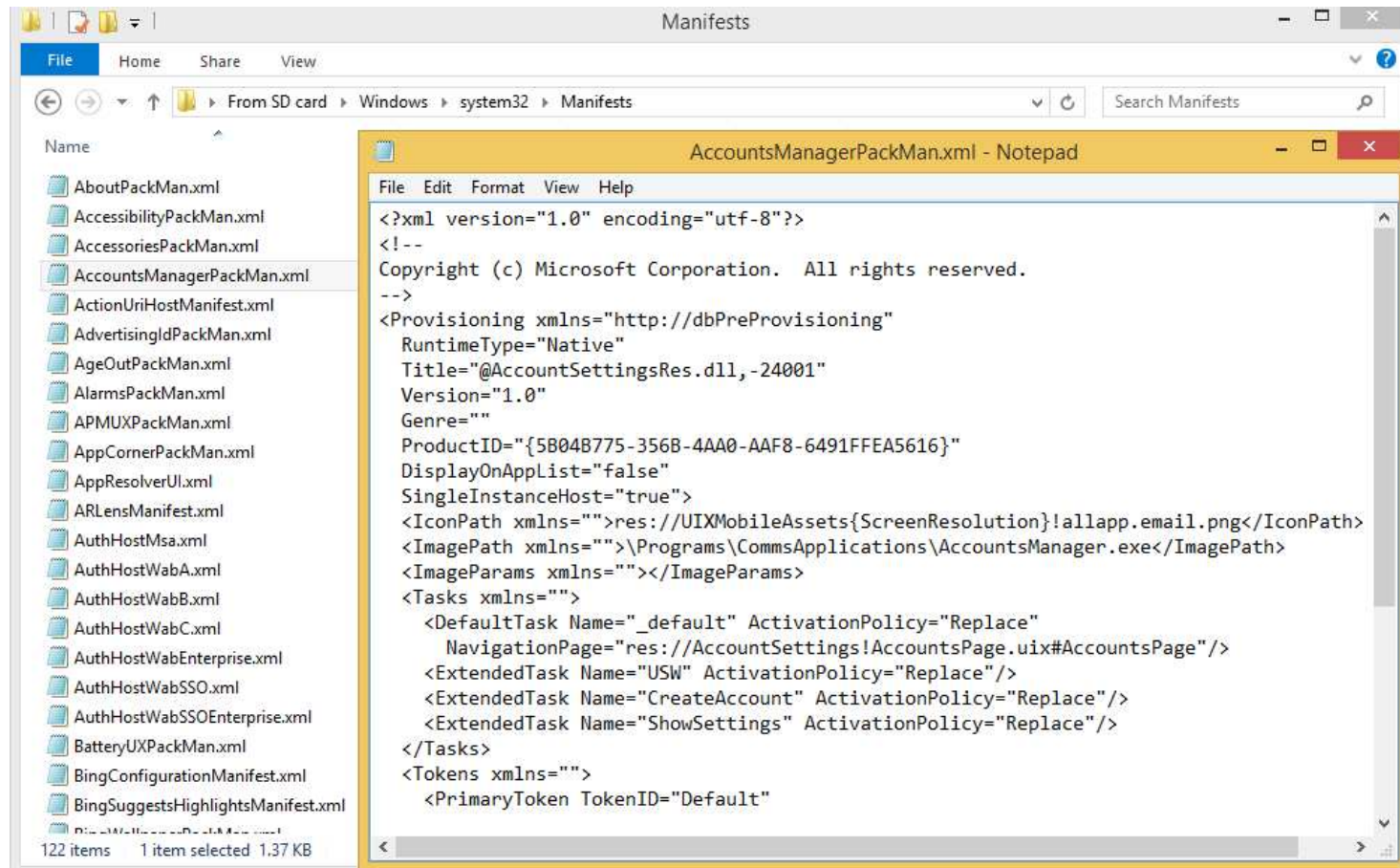


Open Research

Analysis of extracted information



While others are not ...



Agenda



Introduction

Windows Aspects

- ✦ Windows Environment
- ✦ Attack Surface
- ✦ Breaking Out

Mobile Aspects

- ✦ Sandboxing & Encryption

Findings

- ✦ MDM Integration
- ✦ Wi-Fi Sense
- ✦ Low Level Storage API

Conclusion

Windows Phone 8.1

- ✦ Is more similar to iOS and Android than to a Windows desktop
- ✦ Is based on secure and proven good security technologies
- ✦ Is a first step into a more mature Windows 10 ecosystem
- ✦ Is as business ready as your current MDM solution is

Open discussion



Thank you!



Thank you for
your attention!


Contact



Compass Security Deutschland GmbH

Taentzienstr. 18
10789 Berlin
Germany

team@csnc.de | www.csnc.de | +49 30 21 00 253-0

 Secure File Exchange: www.filebox-solution.com

PGP-Fingerprint:

