WIDE ANGLE
INFORMATION SECURITY AND RISK MANAGEMENT

Powered by NTT Com Security

Mobile Device Mismanagement

Vulnerabilities in MDM Solutions and their impact

Stephen Breen
06 AUG 2014

Stephen Breen-Public-Approved

NTT Communications | NTT Com Security

Bios

Stephen Breen

- Senior Consultant

K AS TRUSTED PARTNERS TO:
bilities before anyone else does
ickly — reducing impact on business
ment on the most important challenges

Christopher Camejo

- Director of Assessment Services

Contents

Intro

MDM market

How this started

What we found

What can we do about it

Intro

Everything increases the potential attack surface – even security products

Heartbleed

- Neel Mehta - 2014
- SSL/TLS supposed to protect communication channels
- Vulnerability results in a false sense of security

Antivirus

- Feng Xue - “Attacking Antivirus” - Black Hat Europe 2008
- Vulnerabilities within AV allow full system compromise
- Write malware that gets into the network through the virus scanner

Barracuda

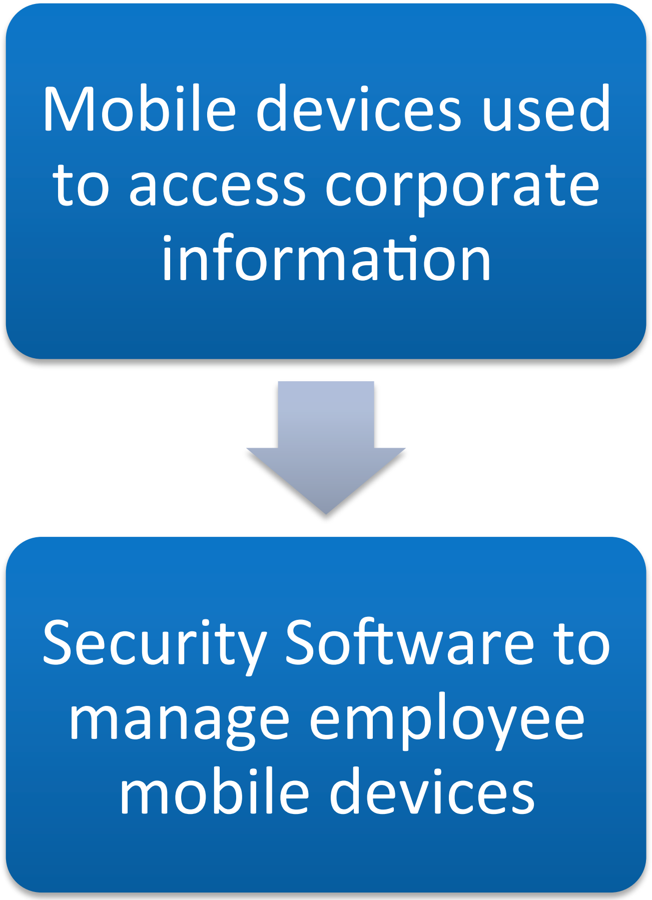
- Stefan Viehböck - 2013
- Vendor hardcoded root backdoor accounts in firewalls, VPNs, etc.
- Your own security products can be turned against you

MDM

- Sebastien Andrivet - “The Security of MDM Systems” - Hack In Paris 2013
- More web interface vulns plus attacks on the device communications

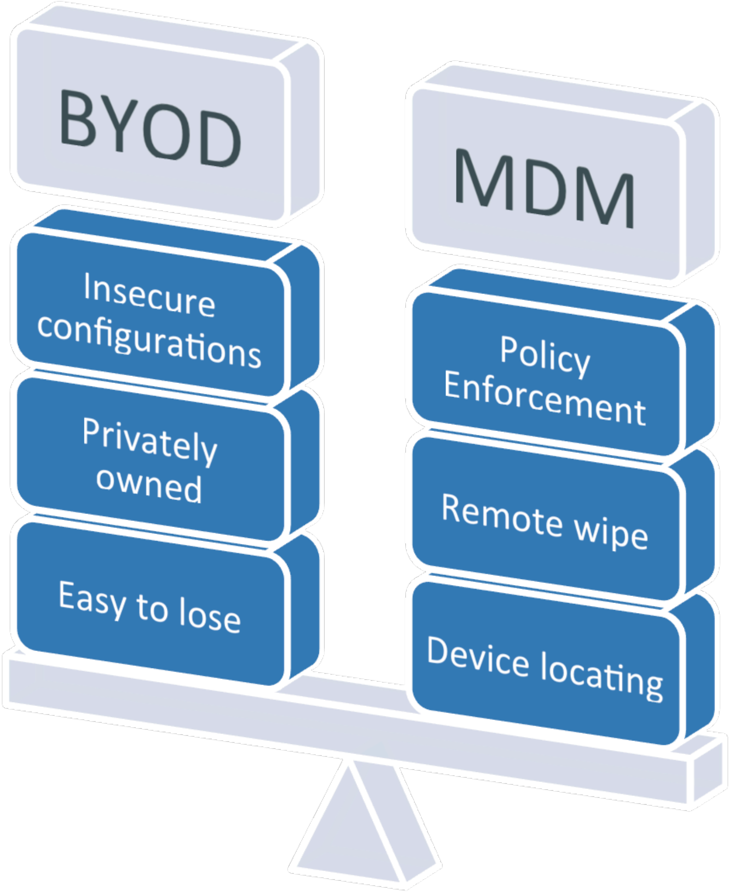
MDM market

What is Mobile Device Management?



Mobile devices used to access corporate information

Security Software to manage employee mobile devices



Deployment Data

Approximately 180 million Enterprise BYOD devices globally

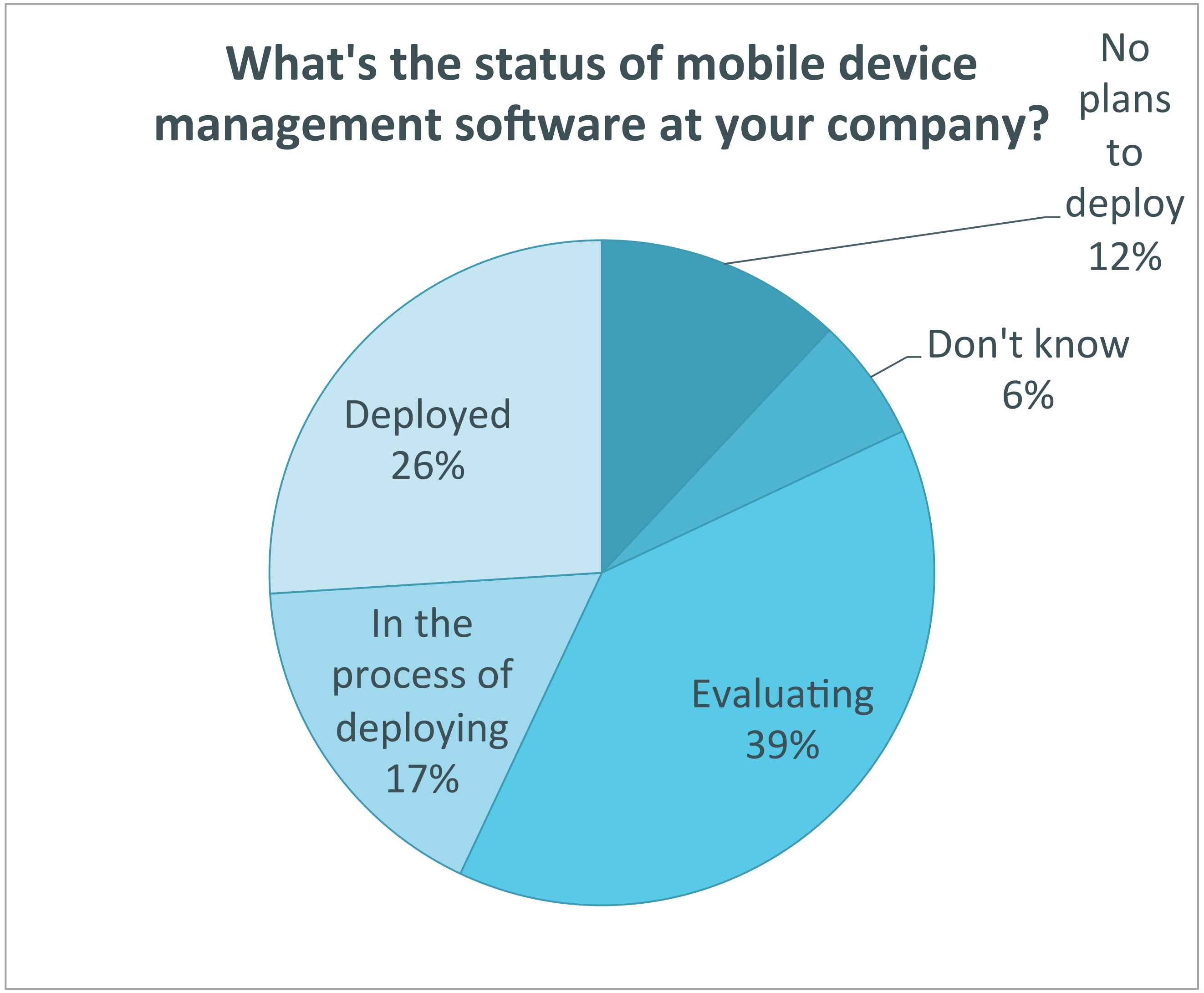
Expected to increase 390 million by 2015.

The U.S. region will lead the market with an estimated 68 percent of the overall market share.

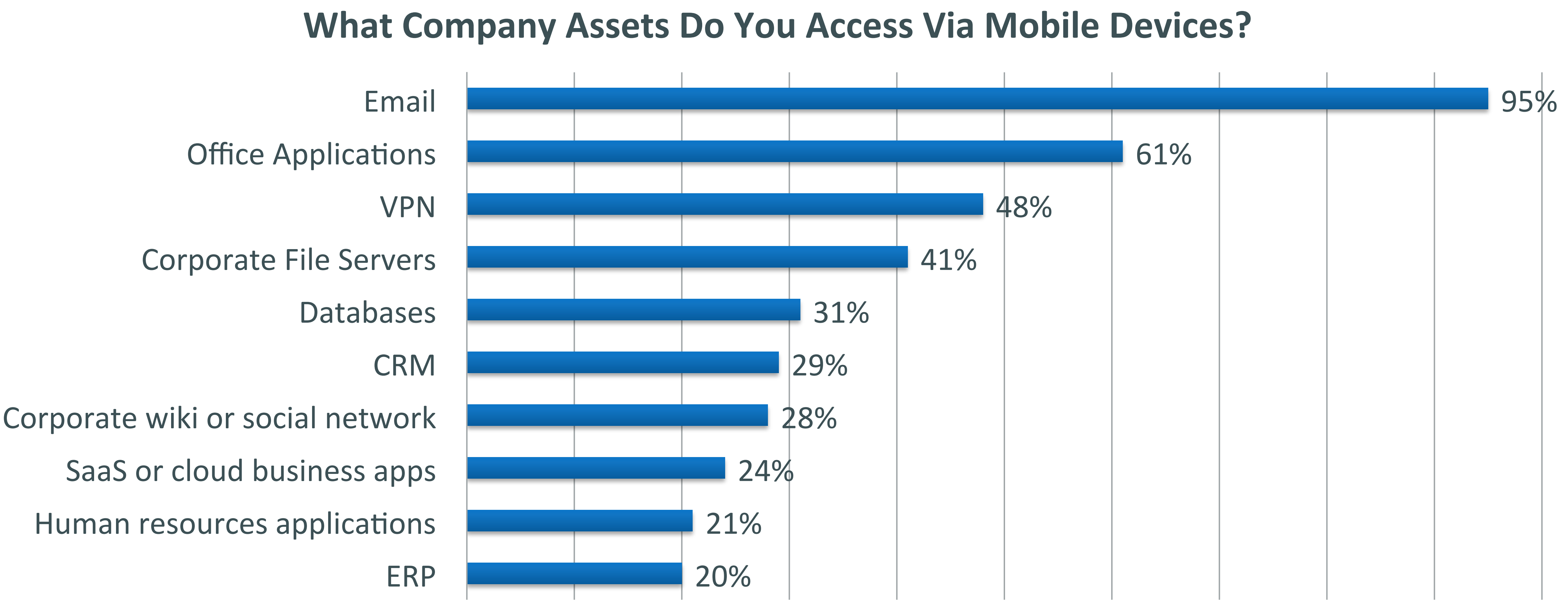
MDM market will grow 23.3% over the next five years.

82% of companies surveyed looking into MDM

- Data: InformationWeek 2013 Mobile Device Management and Security Survey of 307 business technology professionals, September 2012

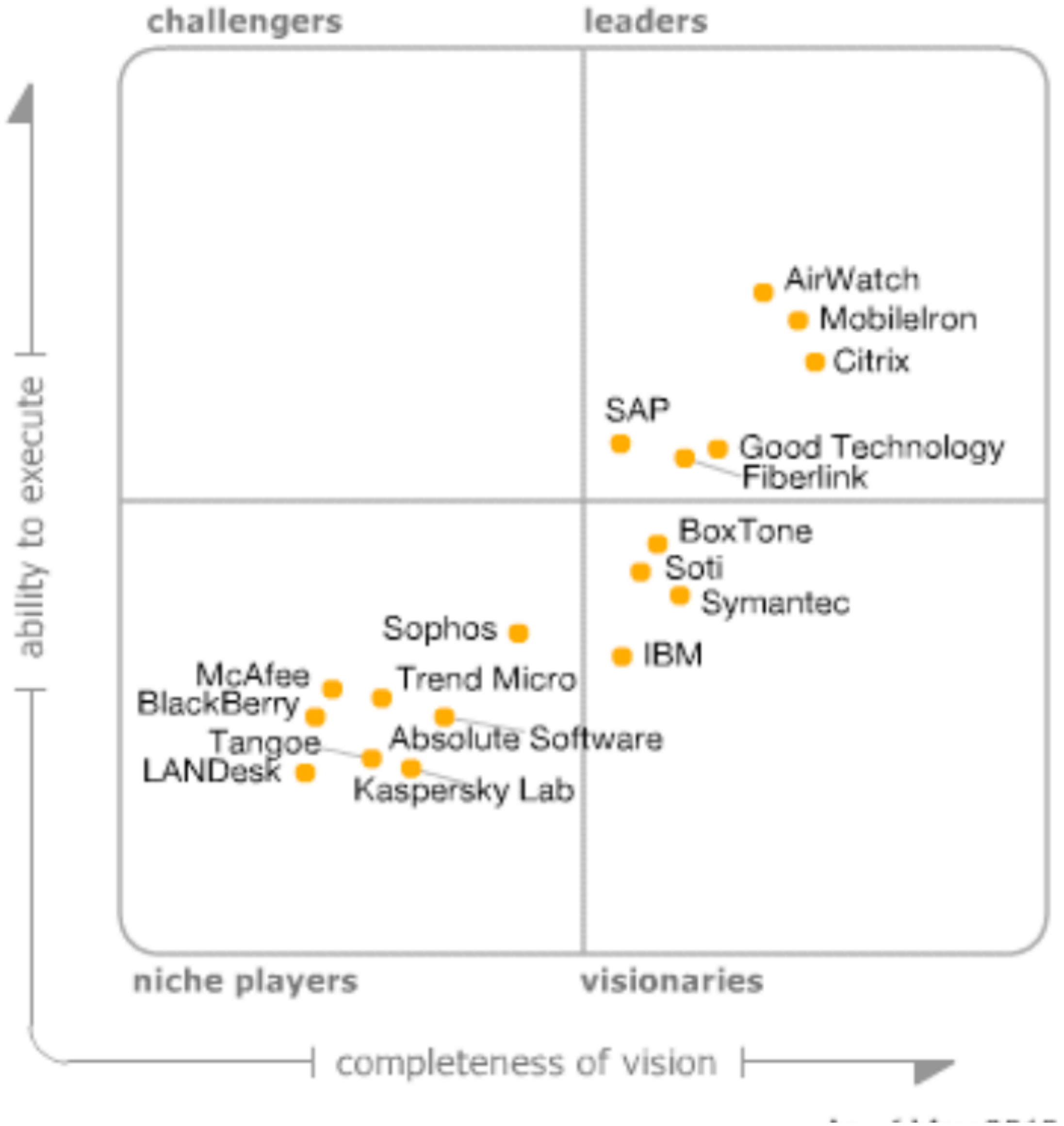


Usage data



Data: InformationWeek 2013 Mobile Device Management and Security Survey of 307 business technology professionals, September 2012

Products



Top-right quadrant: 0 CVE results

- Doesn’t mean there are no vulnerabilities
- Could mean nobody is looking

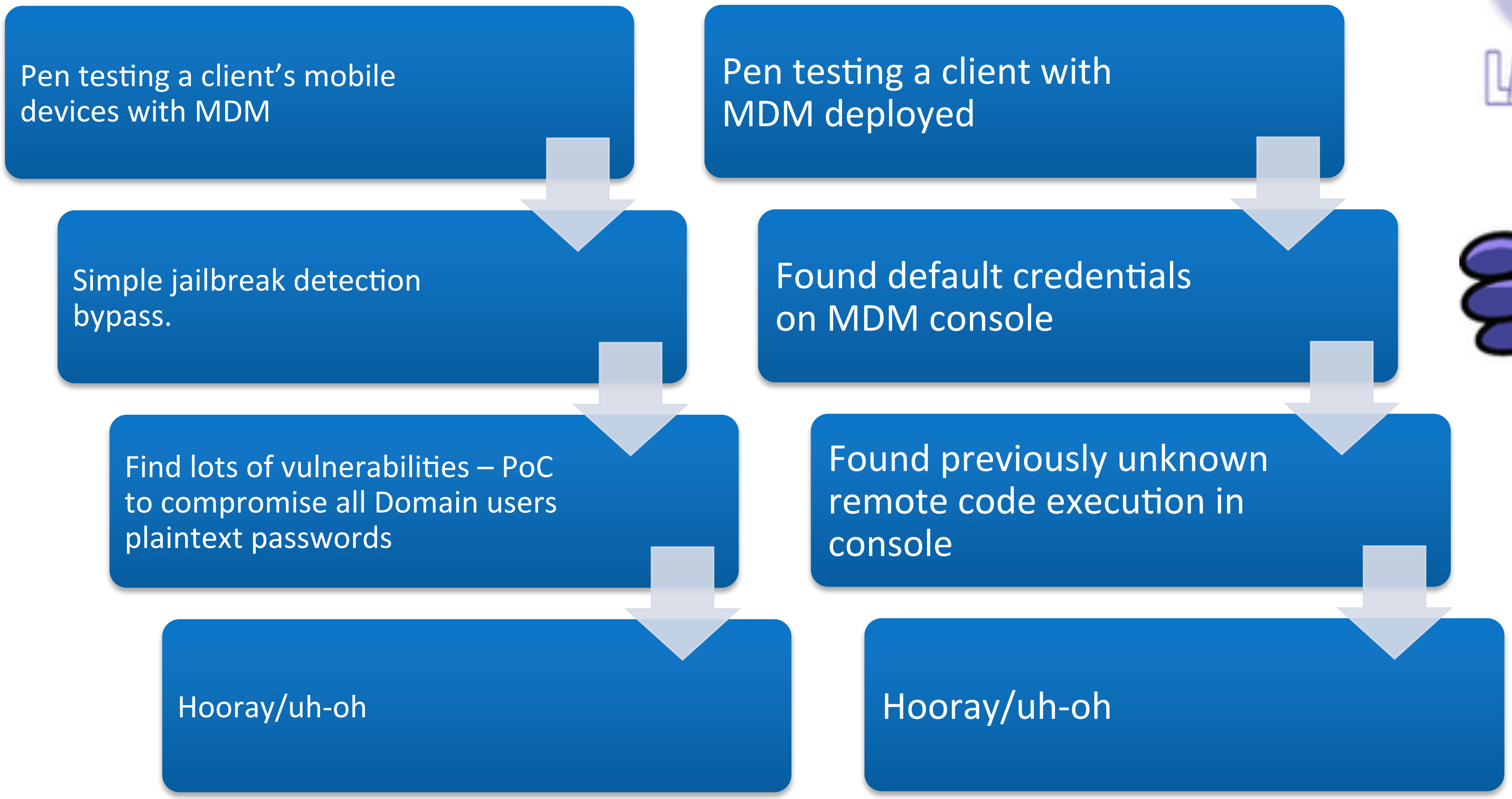
Some products share a common backend

- They likely share common vulnerabilities

Source: Gartner (May 2013)

How this started

The value of a good pen test



Pen testing a client's mobile devices with MDM

Simple jailbreak detection bypass.

Find lots of vulnerabilities – PoC to compromise all Domain users plaintext passwords

Hooray/uh-oh

Pen testing a client with MDM deployed

Found default credentials on MDM console

Found previously unknown remote code execution in console

Hooray/uh-oh

Vendor Relations

Hard to test MDM

- Most vendors don't give out demo products
- Not much tooling or information available to pen testers

Findings disclosed to vendors

- Patches have already been issued and will continue to be issued based on the issues we have identified

What we found

…minus the details

First Glance

We focused on iOS MDM because it uses a standard protocol

- Android's lack of standard does not imply it's better, - just product specific vulnerabilities

iOS enforces an API for MDM

- Most of the code on the mobile device is part of iOS
- The protocol is standardized but the implementations vary
- The server software is also written by the vendor

Vendor code is where vulnerabilities have slipped in

- It's possible to implement reasonably secure MDM on iOS – the protocol seems solid

Android doesn't have an MDM API

- More room for the vendors to make mistakes
- Android implementations may be much worse than iOS

iOS MDM API Enrollment – How it works

Enrollment is the process by which a device becomes managed by MDM

iOS Uses 3 distinct Phases for enrollment:

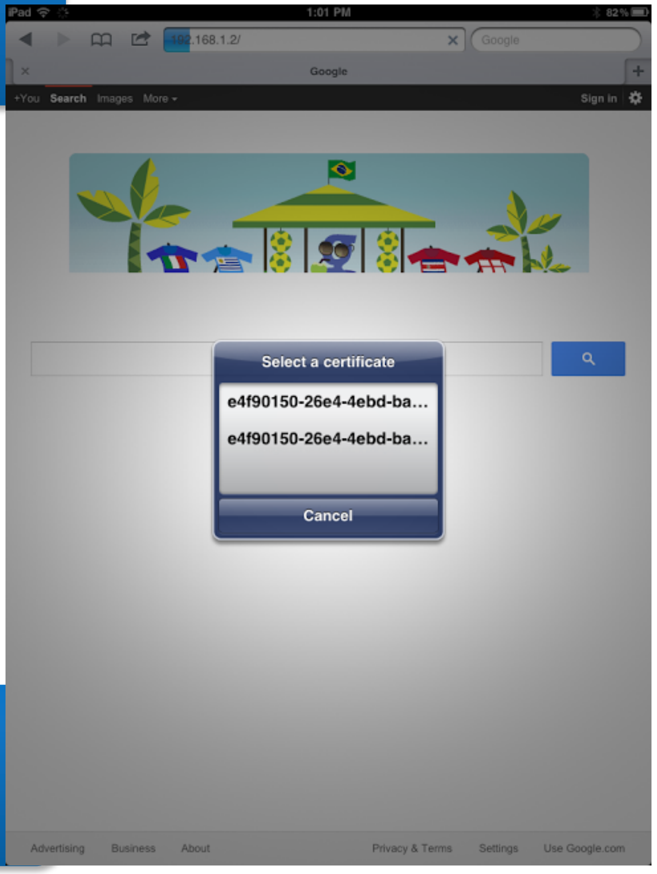
- Authentication – The user authenticates to the MDM server
- Certificate Enrollment – The device and server exchange crypto keys
- Device Configuration – The server applies configuration changes to the device

Typically occurring over HTTP

iOS MDM API Enrollment – Negotiation Issues

Issues:

- Doing enrollment without encrypting communications
- Easily ignored certificate errors
- Predictable tokens
- Tokens remain valid for re-enrollment forever
- Token leakage (external services and improper handling)



Select a certificate

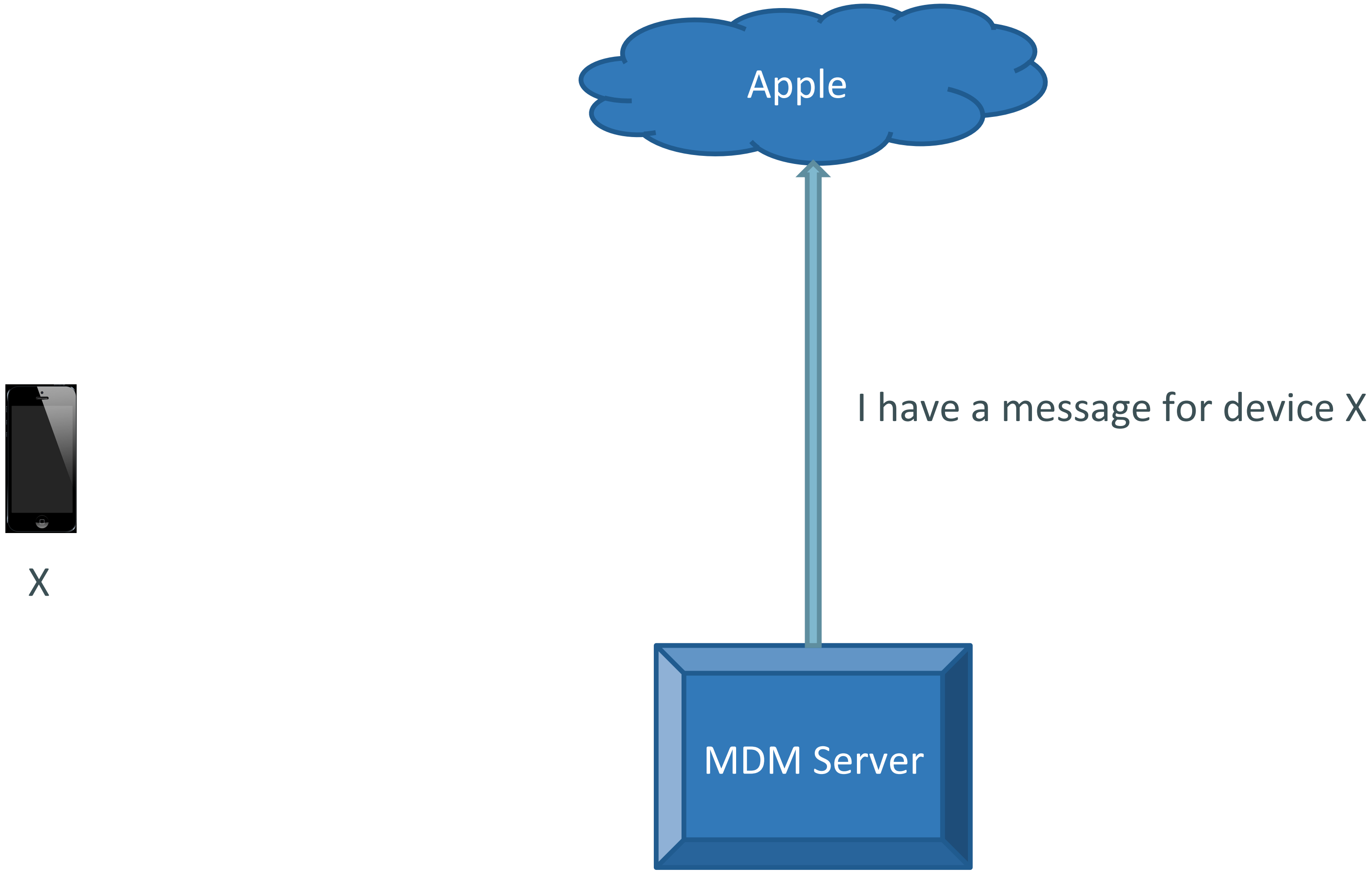
e4f90150-26e4-4ebd-ba...

e4f90150-26e4-4ebd-ba...

Result:

- Compromising tokens results in user impersonation

iOS MDM API Communication – How it works



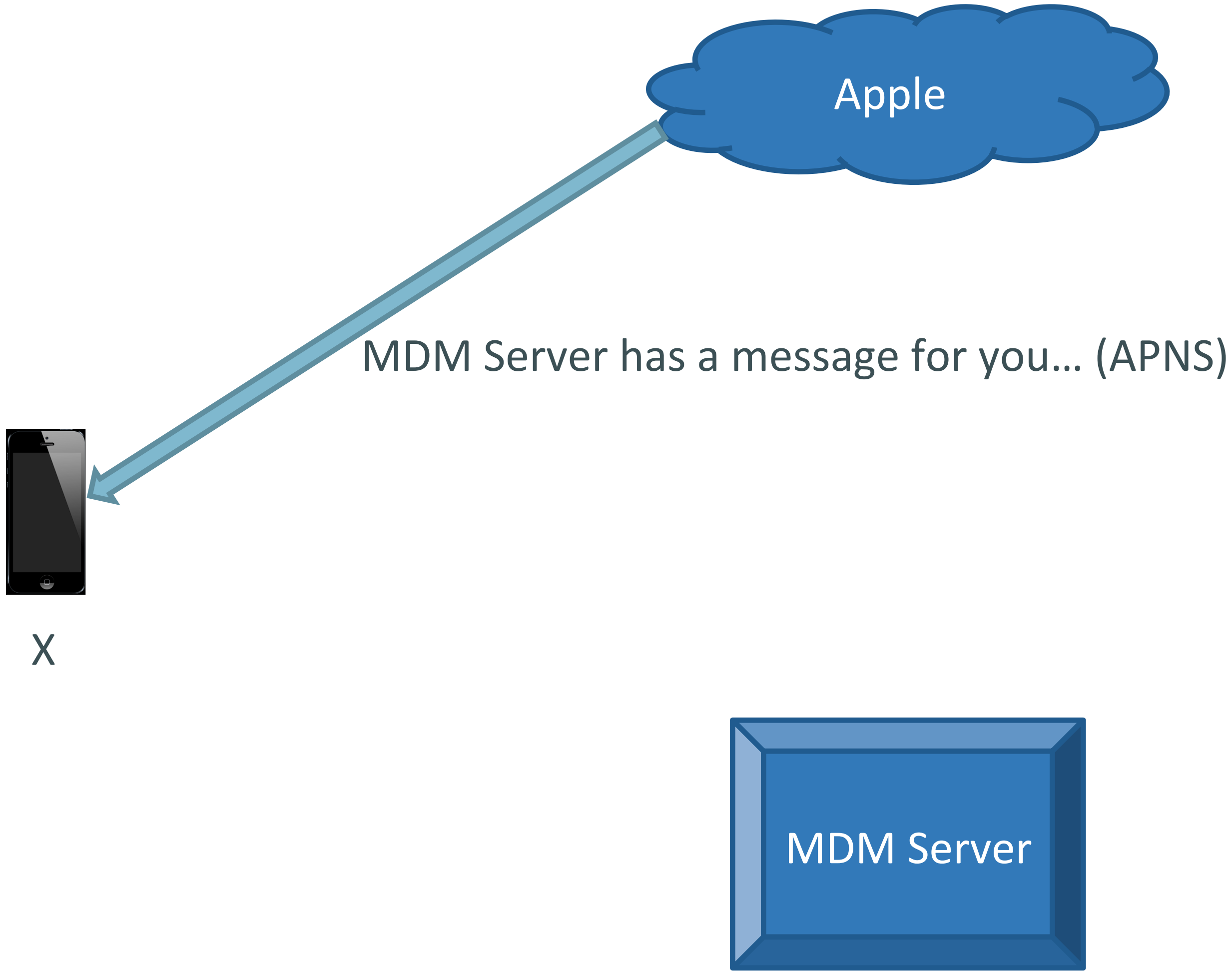
Apple

I have a message for device X

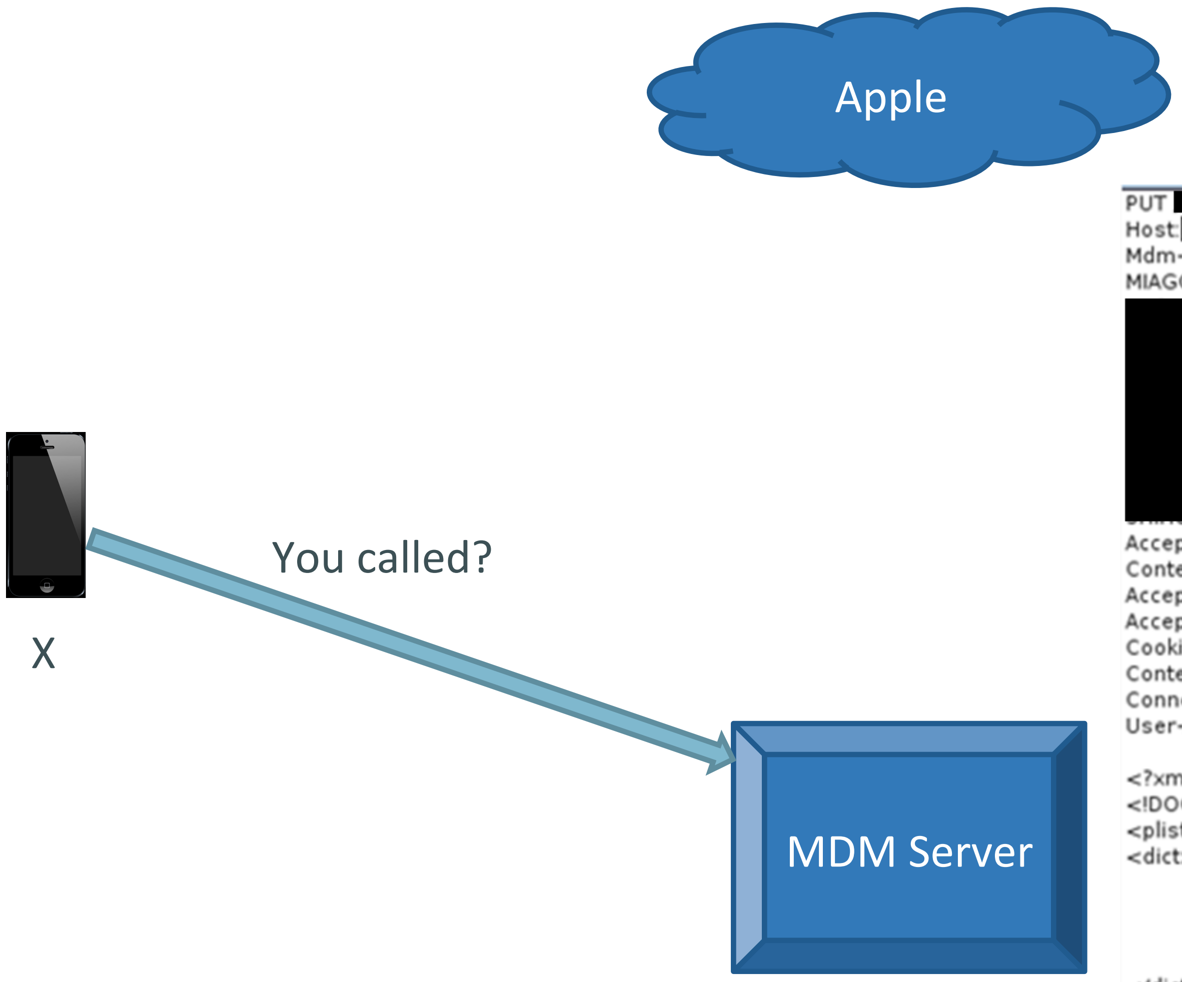
X

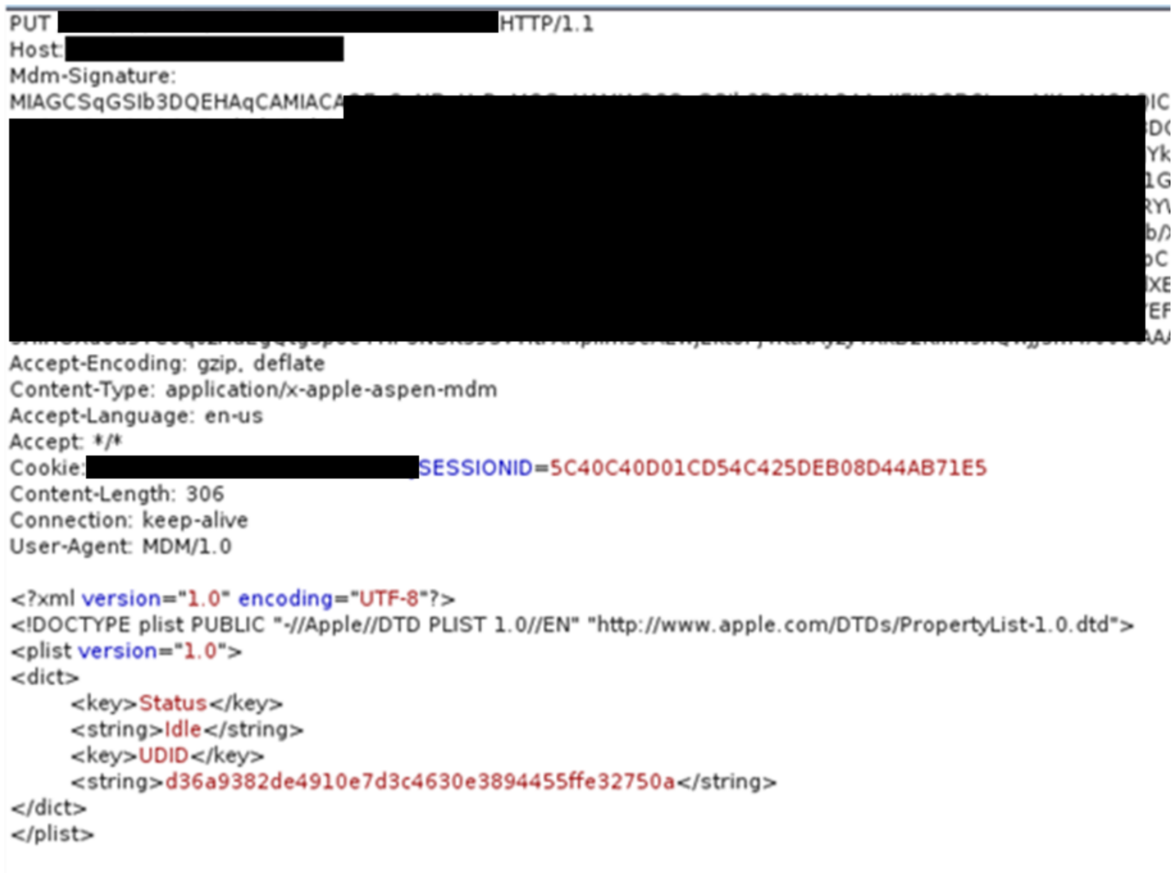
MDM Server

iOS MDM API Communication – How it works



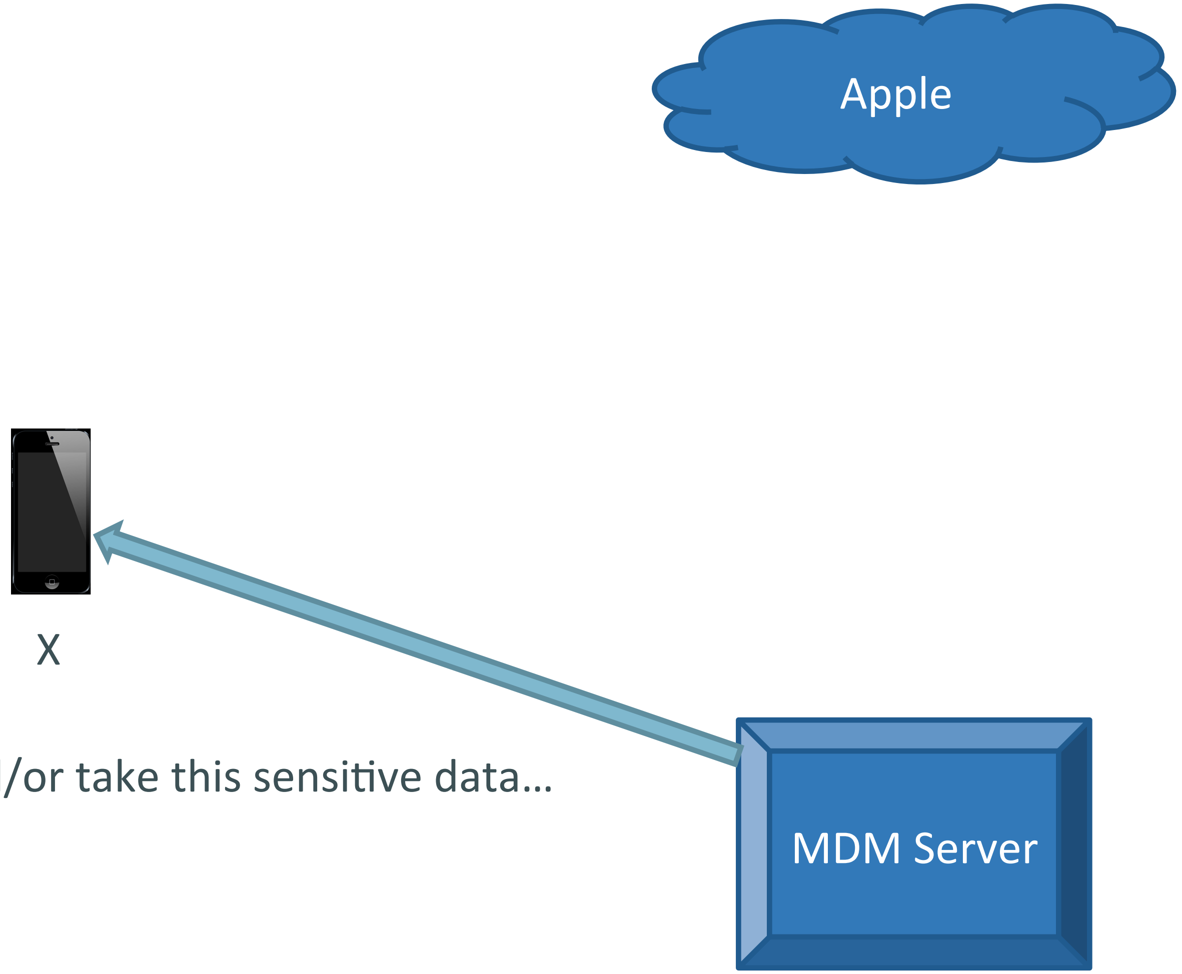
iOS MDM API Communication – How it works





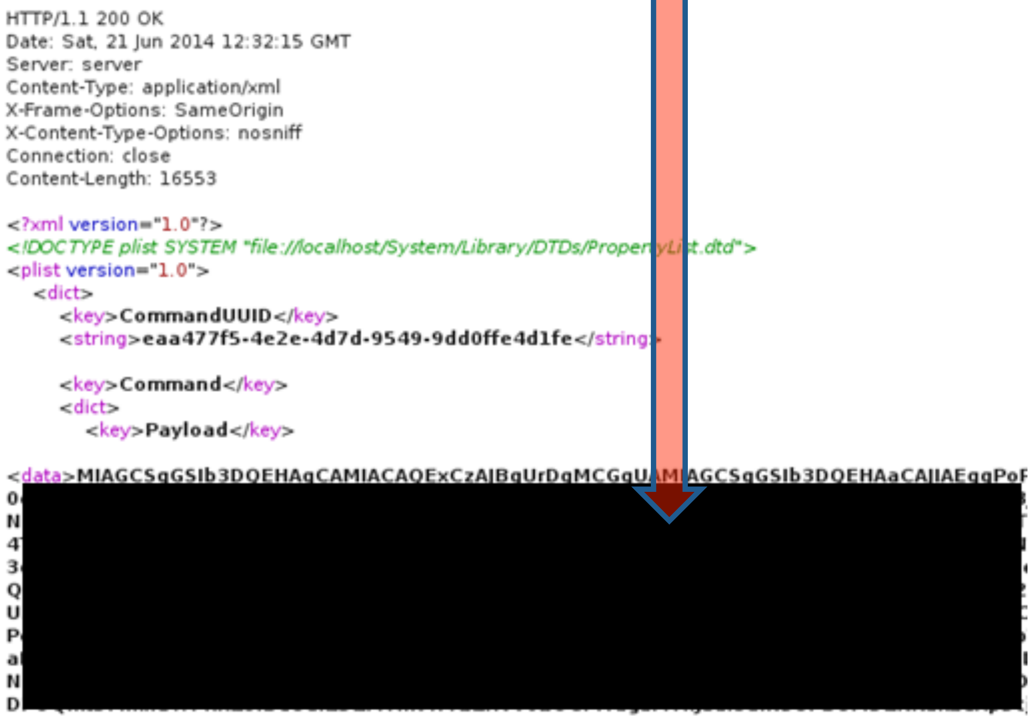
```
PUT [redacted] HTTP/1.1
Host: [redacted]
Mdm-Signature:
MIAGCSqGSIb3DQEHAqCAMIACA[redacted]
Accept-Encoding: gzip, deflate
Content-Type: application/x-apple-aspen-mdm
Accept-Language: en-us
Accept: */*
Cookie: [redacted]SESSIONID=5C40C40D01CD54C425DEB08D44AB71E5
Content-Length: 306
Connection: keep-alive
User-Agent: MDM/1.0

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN" "http://www.apple.com/DTDs/PropertyList-1.0.dtd">
<plist version="1.0">
<dict>
        <key>Status</key>
        <string>Idle</string>
        <key>UDID</key>
        <string>d36a9382de4910e7d3c4630e3894455ffe32750a</string>
</dict>
</plist>
```

iOS MDM API Communication – How it works



Do stuff and/or take this sensitive data…

- Domain Credentials
- WPA2 PSK
- Configuration settings
- …



```
HTTP/1.1 200 OK
Date: Sat, 21 Jun 2014 12:32:15 GMT
Server: server
Content-Type: application/xml
X-Frame-Options: SameOrigin
X-Content-Type-Options: nosniff
Connection: close
Content-Length: 16553

<?xml version="1.0"?>
<!DOCTYPE plist SYSTEM "file://localhost/System/Library/DTDs/PropertyList.dtd">
<plist version="1.0">
  <dict>
    <key>CommandUUID</key>
    <string>eaa477f5-4e2e-4d7d-9549-9dd0ffe4d1fe</string>

    <key>Command</key>
    <dict>
      <key>Payload</key>

<data>MIAGCSqGSIb3DQEHAqCAMIACAQExCzAJBgUrDgMCGgUAMIAGCSqGSIb3DQEHAaCAJIAEggPoF
[redacted]
```

iOS MDM API Communication – Commands

Control	Device Info	Configuration	Device -> Server
Lock	List Profiles	Install Profile	Token Update
Clear Passcode	Installed Applications	Remove Profile	Authenticate
Wipe	Certificate List	Install Application	CheckOut
	Provisioning Profiles	Remove Application	Status
	Restrictions	Settings	
	Managed Applications	Install Provisioning Profile	
	Security Information	Remove Provisioning Profile	

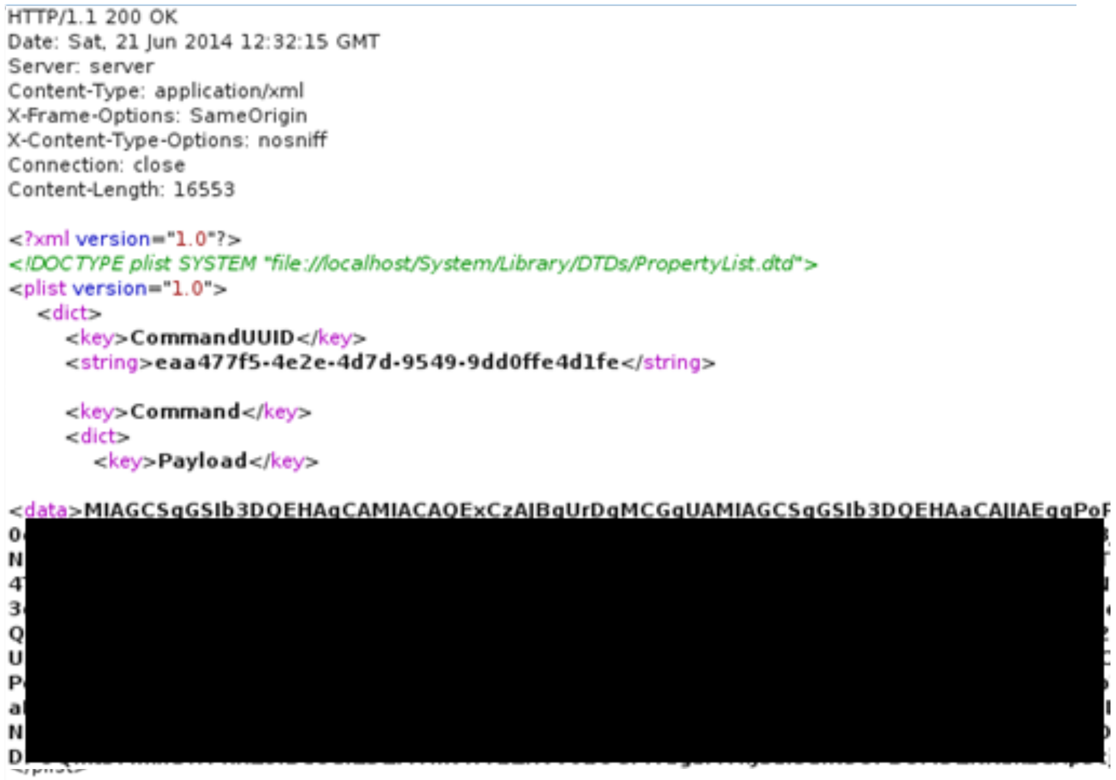
iOS MDM API Communications – Negotiation Issues

MDM-Signature not available in some products

- Send fake messages on behalf of devices
- DoS MDM service by changing tokens
- Tell server devices don’t want to be enrolled anymore
- Trick server into issuing wipe commands
- Steal profile data (AD credentials, WPA keys, etc.)

Payload encryption disabled in some products

- Can remotely intercept sensitive data going from the server to the device
- Domain credentials (plaintext?!), WPA2 pre-shared keys, other sensitive configuration information…



```
HTTP/1.1 200 OK
Date: Sat, 21 Jun 2014 12:32:15 GMT
Server: server
Content-Type: application/xml
X-Frame-Options: SameOrigin
X-Content-Type-Options: nosniff
Connection: close
Content-Length: 16553

<?xml version="1.0"?>
<!DOCTYPE plist SYSTEM "file://localhost/System/Library/DTDs/PropertyList.dtd">
<plist version="1.0">
  <dict>
    <key>CommandUUID</key>
    <string>eaa477f5-4e2e-4d7d-9549-9dd0ffe4d1fe</string>

    <key>Command</key>
    <dict>
      <key>Payload</key>

<data>MIAGCSqGSIb3DQEHAqCAMIACAQExCzAJBgUrDgMCGgUAMIAGCSqGSIb3DQEHAaCAJIAEggPoF
[redacted]
```

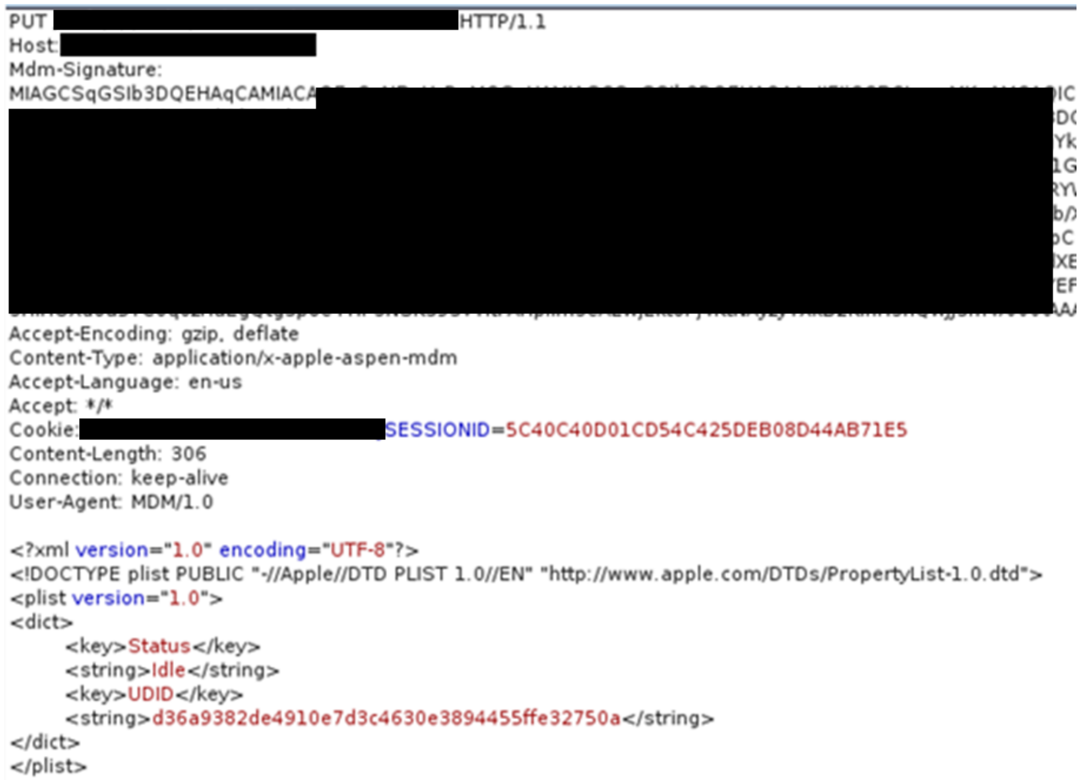
iOS MDM API Communications – Negotiation Issues

Injection Flaws

- SQLi
- XXE
- We were able to create a BURP extension to automatically generate spoofed MDM-Signature headers

Flawed Signature Validation

- Not all signature validation methods are created equal
- Some products may not link keys to users
- Some products may not check issuing CA



```
PUT [redacted] HTTP/1.1
Host: [redacted]
Mdm-Signature:
MIAGCSqGSIb3DQEHAqCAMIACA[redacted]
Accept-Encoding: gzip, deflate
Content-Type: application/x-apple-aspen-mdm
Accept-Language: en-us
Accept: */*
Cookie: [redacted] SESSIONID=5C40C40D01CD54C425DEB08D44AB71E5
Content-Length: 306
Connection: keep-alive
User-Agent: MDM/1.0

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN" "http://www.apple.com/DTDs/PropertyList-1.0.dtd">
<plist version="1.0">
<dict>
        <key>Status</key>
        <string>Idle</string>
        <key>UDID</key>
        <string>d36a9382de4910e7d3c4630e3894455ffe32750a</string>
</dict>
</plist>
```

What does this mean?

For Users:

Everything increases attack surface, even security products

Don’t deploy anything unless there’s a business need

- “Everybody else is doing it” isn’t a business need.

Due diligence (e.g. pen testing) of products before you choose and deploy

- When was the last time you had somebody look for zero-day vulnerabilities in a software product you bought?

Proper care and feeding of things you’ve deployed

- Hardened configuration
- Vulnerability management program
- Monitoring logs and alerts for suspicious activity

For Users:

Real pen testing

- More than vulnerability scanning
- APT are looking for zero-days, you should too
- Keep in mind this all started at a client during a routine pen test

Look at risk across the organization

- Security isn't about throwing more fancy boxes on the network, those are just tools
- In order for tools to be effective they need to be deployed appropriately and have operators who know how to use them (and have the time)
- If you don't know where your risk is you can't deploy tools appropriately

For Product Vendors:

Software Development LifeCycle

- Everything is webified so your devs better eat/breathe/sleep OWASP
- Pen test your own products (before somebody does it for you)
- Your customers shouldn't be your QA team
- If your QA team doesn't know how to find vulnerabilities then find somebody who can

Don't rely on security by obscurity

- We can reverse-engineer your protocol faster than you wrote it
- So can the bad guys

Authenticate all the things

- Certificates, tokens, signatures, and encryption exist for a reason, use them
- If you're making your own version of any of those: you're doing it wrong

Q&A

No, we won’t name vendors

- There are patches for many of these issues but people need time to apply them
- And some of these issues may still be unpatched
- But we would be happy to pen test your MDM deployment ☺

Stephen Breen

- Senior Security Consultant
- NTT Com Security
- stephen.breen@nttcomsecurity.com