MICROSOFT VULNERABILITY RESEARCH

How to be a finder as a vendor

WHO ARE THESE FINE GENTLEMEN

- David Seidman
  - Manager of MSVR Program
  - Likes authentication, hates passwords

- Jeremy Brown
  - MSVR Contributor since 2011
  - Likes bugs, but also likes making things more secure

AGENDA

- What is Microsoft Vulnerability Research?
- The MSVR Process
  - How it works
  - And how things can go wrong

AGENDA

- Case Studies
  - Libavcodec
  - Comodo GeekBuddy
  - VMware Player
  - Blackberry “Print To Go”
- Lessons Learned

WHAT WE'RE NOT COVERING

- How Microsoft handles vulnerabilities in 3rd party software distributed with our products
- Any information about MSVR bugs **in the queue** for public release
- The ethics of disclosure or debating which philosophy is the greatest

WHAT IS MICROSOFT VULNERABILITY RESEARCH?

ORIGINS

- MSVR started in 2008
    - Founded by Katie Moussouris
    - Announced at the BlackHat conference

ORIGINS

- MSRC cases and internal finds were affecting many other vendors
- We needed a way to coordinate with vendors across the industry in order to ensure fixes for these bugs materialize

MSVR ISN'T

- MSRC
  - Microsoft Security Response Center
  - Handles security incidents and vulnerabilities affecting **Microsoft products**

- Microsoft Bounty Programs
  - Cash for defensive ideas and IE11 Preview bugs

MSVR ISN'T

- HackerOne
  - Hosts of the Internet Bug Bounty program
  - “Rewards friendly hackers who contribute to a more secure internet”
  - Sponsored by both Microsoft and Facebook

MSVR IS…

- A program to help Microsoft employees report security vulnerabilities to third party software vendors
- Provide assistance to finders
  - People to answer questions and ping the vendor
  - Security contact database
  - The resources to find contacts if no public ones exist

MSVR IS…

- Objectives
  - Prevent miscommunication
  - Keep all parties informed
  - Provide transparency for both sides

MSVR ADVISORIES

- Dedicated Microsoft webspace to display and archive vulnerability and fix information
  - http://technet.microsoft.com/en-us/security/msvr
- Each advisory credits the researcher for the find
  - Unless you want to be anonymous, of course

WHY THE FOCUS ON THIRD PARTY

- Windows runs lots of third-party code. That code becomes attack surface for Microsoft users.
  - Adobe Reader and Oracle Java account for the majority exploits used to compromise PCs
- Not just PC software
  - Routers in our datacenters
  - Firmware in our devices
  - Apps in our software stores

Reference: http://download.microsoft.com/download/5/0/3/50310CCE-8AF5-4FB4-83E2-03F1DA92F33C/Microsoft_Security_Intelligence_Report_Volume_15_English.pdf

WHY THE FOCUS ON THIRD PARTY

- Often the vulnerabilities affect Microsoft too
  - Protocol flaws
    - DNS
    - SSL
  - Common coding and design flaws

SECURING THE ECOSYSTEM

- Here’s a short list of vendors we’ve worked with at MSVR

- Adobe, AOL, Apple, Blackberry, CA, Cisco, Citibank, Comodo, Fidelity, Google, Hex-Rays, HP, IBM, Intel, Intuit, Lenovo, Mozilla, Nullsoft, Nvidia, OpenOffice, Opera, Oracle, PGP, RealNetworks, SAP, Symantec, VMware, Wireshark, WordPress, Yahoo!

- ….as well as many, many more

GOALS

- Ensure that Microsoft works with others the same way we'd like them to work with us
  - Coordinated vulnerability disclosure so that Microsoft employees do not drop 0-days
  - Reproducible and interesting bugs
  - Good repro and explanation

Reference: http://blogs.technet.com/b/msrc/archive/2011/04/19/coordinated-vulnerability-disclosure-from-philosophy-to-practice.aspx

GOALS

- Help Microsoft finders out
  - Make sure bugs get fixed
  - Release advisories
- Help secure the Microsoft ecosystem
- Build relationships with other vendors

WHO ARE FINDERS?

- Individual Microsoft employees who find security bugs for various reasons
  - Hobby
  - Securing software they use
- Product groups working extensively with a third party product
  - E.g. Office finding Adobe Reader bugs when testing Word’s Save as PDF function
  - Often many bugs are discovered at once, or a stream of bugs is generated on an ongoing basis
- Product groups hitting one-off bugs
  - It is not uncommon to hit a bug in a third-party component while just testing functionality

WHICH VULNERABILITIES ARE ELIGIBLE?

- Found by a Microsoft employee
  - Whether found on own time or otherwise, using company resources or not
- Critical and Important on SDL Bug Bar
  - Remote code execution, server DoS, XSS, SQLi, MITM, a few others
- Affects a product on a Microsoft platform or used in a Microsoft datacenter
  - E.g. iPhone apps are not eligible
- These aren’t hard rules – designed to ensure high ROI

MSVR REQUIREMENTS

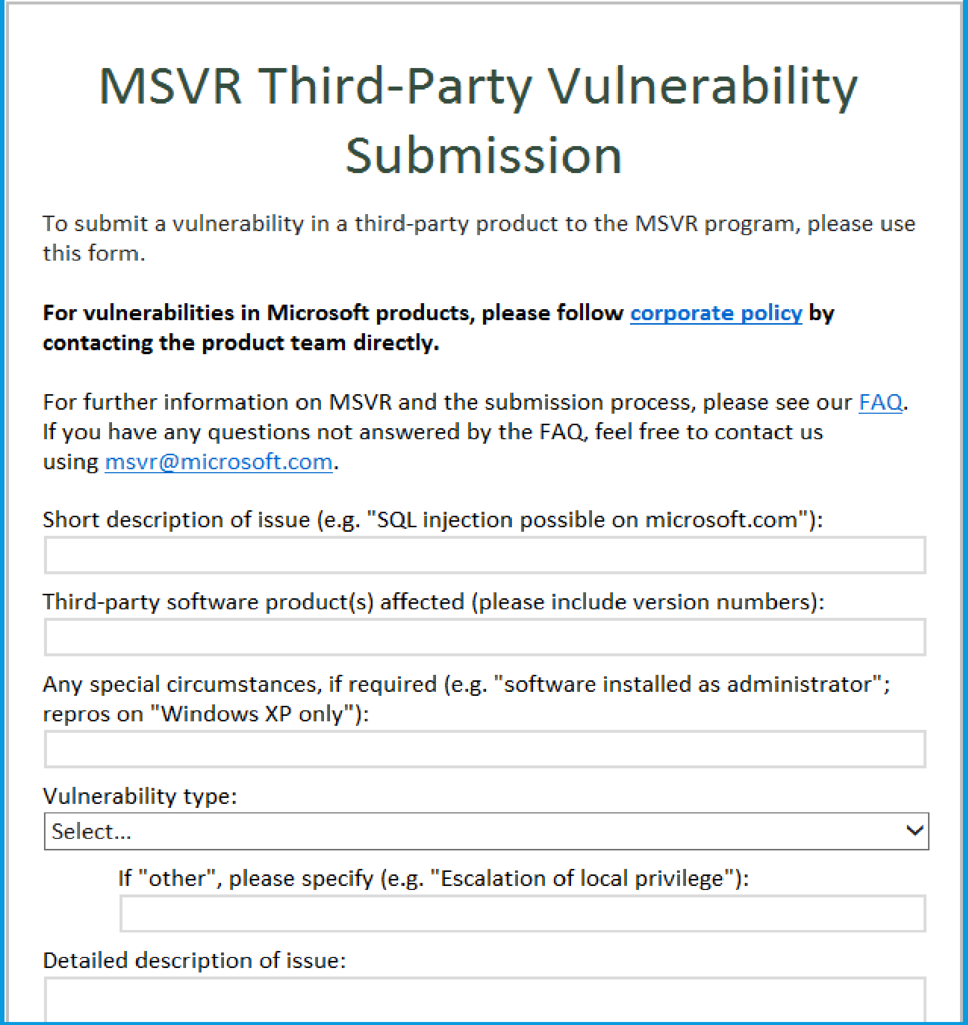
- I am not a lawyer, so this is a paraphrase of the actual policy
- Microsoft employees must use CVD under all circumstances
  - CVD: Coordinated Vulnerability Disclosure (the new one, not “responsible disclosure”)
  - =no odays per Microsoft’s policy
- Employees must notify MSVR of all vulnerabilities they report
  - Exception: existing working/support/partnership relationships can continue
  - Using MSVR to manage the process is optional for bugs found on personal time

MSVR REQUIREMENTS

- Third-party bugs found outside company time and not using company assets may be reported through a vuln broker using CVD
  - The employee can keep the money
  - This includes bug bounties too

THE MSVR PROCESS

STEP 1: REPORT VULNERABILITY



MSVR Third-Party Vulnerability Submission

To submit a vulnerability in a third-party product to the MSVR program, please use this form.

For vulnerabilities in Microsoft products, please follow corporate policy by contacting the product team directly.

For further information on MSVR and the submission process, please see our FAQ. If you have any questions not answered by the FAQ, feel free to contact us using msvr@microsoft.com.

Short description of issue (e.g. "SQL injection possible on microsoft.com"):

Third-party software product(s) affected (please include version numbers):

Any special circumstances, if required (e.g. "software installed as administrator"; repros on "Windows XP only"):

Vulnerability type:

If "other", please specify (e.g. "Escalation of local privilege"):

Detailed description of issue:

STEP 1 MISFIRE: CLASSIC 0-DAY

- <insert any Windows oday full disclosure post here in the last 20 years>

STEP 2: ENSURE QUALITY

- MSVR ensures that all required elements are present:
  - Qualifying bug details
  - Proof of concept file or solid repro steps
  - Description of issue, including affected products and versions, severity, etc.
  - Stack trace
  - Ideas for workarounds or code fixes
- We'll go back-and-forth with finders until it meets quality bar
- Won't ship if it doesn't

STEP 2 MISFIRE: NOT A BUG

- When logging into Windows
  - If you have the number 8 in your login password, and
  - You have NumLock off and
    - You use the number pad when typing the number 8
    - You will switch focus to the username field and might accidentally type the rest of your password into the username field

STEP 3: CHECK FOR MICROSOFT IMPACT

- Does Microsoft have code that could be similarly affected?
  - Does an SSL bug affect our SSL stack?
  - Does a browser bug affect Internet Explorer?
  - Etc.
- If so, coordinate with third parties to align their fix schedule with ours

STEP 3 MISFIRE: WE 0-DAY OURSELVES

- Microsoft researchers: Online ad networks' payment processing can be theoretically exploited for fraud!
- Just like Bing's
- Researchers: "We thought it would be okay because we didn't mention Bing"

STEP 4: REPORT VULNERABILITY

- Find the vendor's security contact point (email, web form, etc) if we don't already have it
  - If they don't have one, we try harder ☺
- Tell them we have a vulnerability to report and request PGP or S/MIME key
  - Perhaps explain to them what PGP is…
- Encrypt and send details

STEP 4 MISFIRE: SALES PURGATORY

- Vendor: What’s your customer ID?
- Microsoft: We don’t have a customer ID, we found a security problem with your website.
- Vendor: Oh, well with no customer ID we can’t help you. Would you like to buy our product?
- Microsoft: We don’t want help or to buy your product. We’re trying to help you.
- Vendor: Thank you for contacting Vendor. Your email is very important to us.

STEP 5: MONITOR

- Follow up with company and internal finder to track their fix through release
- Resolve questions about repro and severity
- Vendor may send a private, fixed version for the finder to confirm the bug is fixed
- Keep all parties up to date with plans for updates, blog post, conference presentations, etc.

STEP 5 MISFIRE: SURPRISE!

- Oh that bug? We patched that six months ago.

STEP 6: SHIP UPDATE

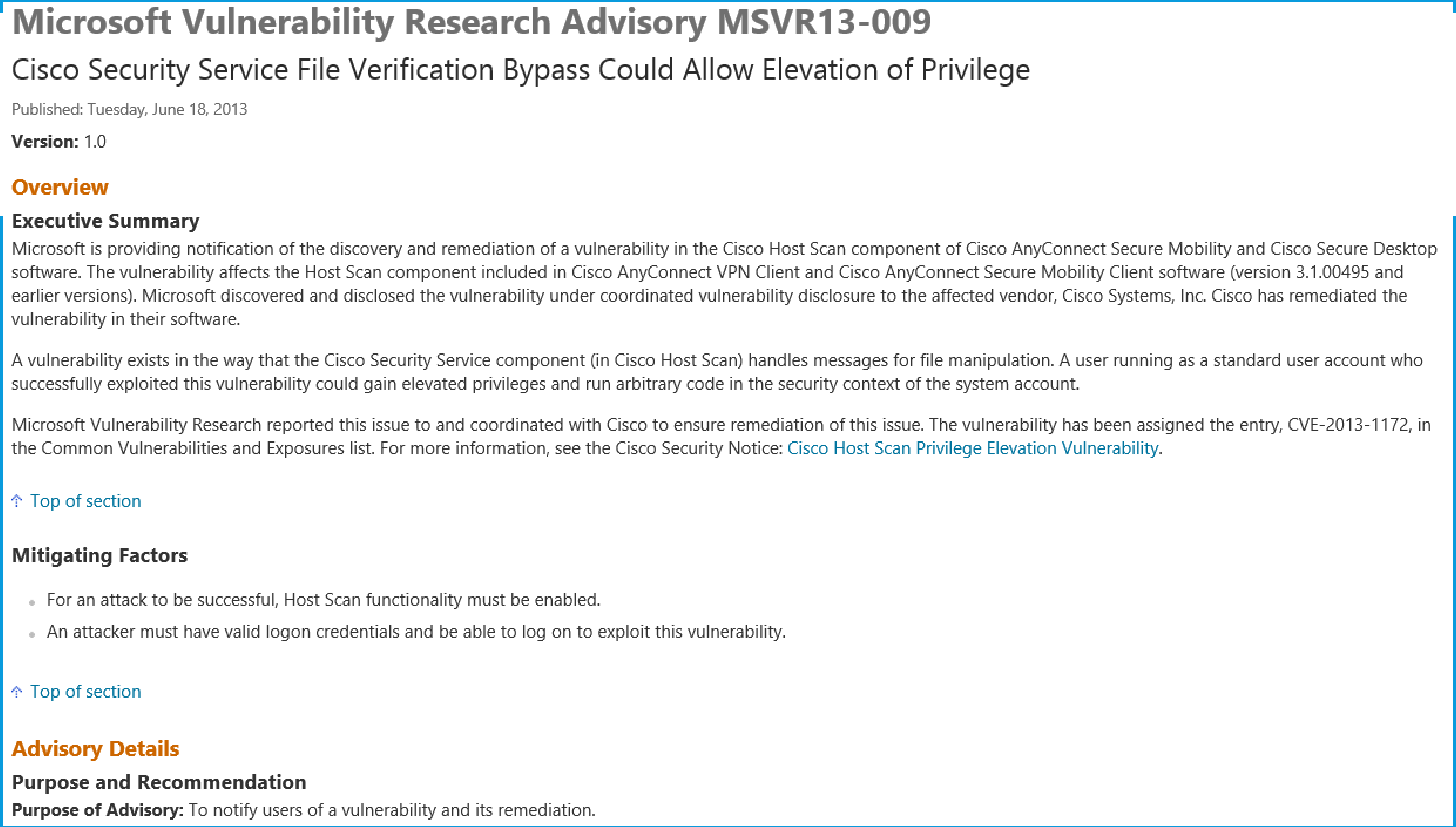
- Vendor releases update
  - Implore them to credit our researcher
- If they “forget”, we’ll ping them and recommend it again ☺

STEP 6 MISFIRE: NO CREDIT

- Vendor: Here’s the fix! <no credit to finder>
- Finder: Hey!

STEP 7: MSVR ADVISORY

- Released when we think a bug particularly merits Microsoft customers' attention
  - Optional
  - Not all vulnerabilities get advisories
- Released with or (typically) after the vendor releases a patch
  - In case of active attacks, we could release one proactively, but we have yet to do so
- Purpose is to notify our customers of the patch and remind them to install it
- Finder always has the option to release their own advisory in coordination with MSVR once vendor has patched



Microsoft Vulnerability Research Advisory MSVR13-009

Cisco Security Service File Verification Bypass Could Allow Elevation of Privilege

Published: Tuesday, June 18, 2013

Version: 1.0

Overview

Executive Summary

Microsoft is providing notification of the discovery and remediation of a vulnerability in the Cisco Host Scan component of Cisco AnyConnect Secure Mobility and Cisco Secure Desktop software. The vulnerability affects the Host Scan component included in Cisco AnyConnect VPN Client and Cisco AnyConnect Secure Mobility Client software (version 3.1.00495 and earlier versions). Microsoft discovered and disclosed the vulnerability under coordinated vulnerability disclosure to the affected vendor, Cisco Systems, Inc. Cisco has remediated the vulnerability in their software.

A vulnerability exists in the way that the Cisco Security Service component (in Cisco Host Scan) handles messages for file manipulation. A user running as a standard user account who successfully exploited this vulnerability could gain elevated privileges and run arbitrary code in the security context of the system account.

Microsoft Vulnerability Research reported this issue to and coordinated with Cisco to ensure remediation of this issue. The vulnerability has been assigned the entry, CVE-2013-1172, in the Common Vulnerabilities and Exposures list. For more information, see the Cisco Security Notice: Cisco Host Scan Privilege Elevation Vulnerability.

↑ Top of section

Mitigating Factors

- For an attack to be successful, Host Scan functionality must be enabled.
- An attacker must have valid logon credentials and be able to log on to exploit this vulnerability.

↑ Top of section

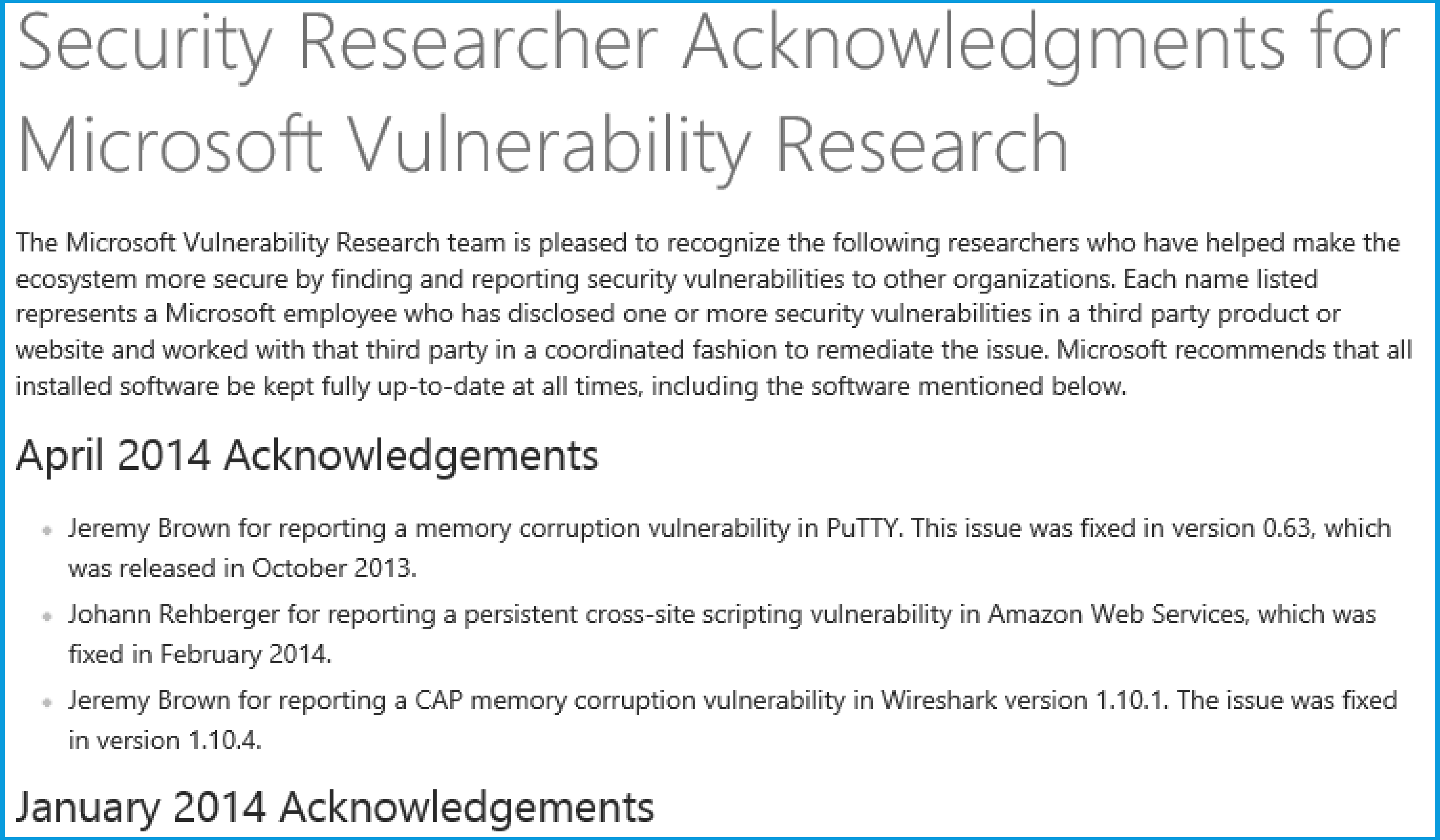
Advisory Details

Purpose and Recommendation

Purpose of Advisory: To notify users of a vulnerability and its remediation.

STEP 7B: MSVR CREDITS

- When we don't do a full advisory, still provide internal finders credit



Security Researcher Acknowledgments for Microsoft Vulnerability Research

The Microsoft Vulnerability Research team is pleased to recognize the following researchers who have helped make the ecosystem more secure by finding and reporting security vulnerabilities to other organizations. Each name listed represents a Microsoft employee who has disclosed one or more security vulnerabilities in a third party product or website and worked with that third party in a coordinated fashion to remediate the issue. Microsoft recommends that all installed software be kept fully up-to-date at all times, including the software mentioned below.

April 2014 Acknowledgements

- Jeremy Brown for reporting a memory corruption vulnerability in PuTTY. This issue was fixed in version 0.63, which was released in October 2013.
- Johann Rehberger for reporting a persistent cross-site scripting vulnerability in Amazon Web Services, which was fixed in February 2014.
- Jeremy Brown for reporting a CAP memory corruption vulnerability in Wireshark version 1.10.1. The issue was fixed in version 1.10.4.

January 2014 Acknowledgements

CASE STUDIES

CASE STUDY: LIBAVCODEC

- MSVR12-017
  - Vulnerabilities in FFmpeg Libavcodec Could Allow Arbitrary Code Execution

- Fuzzing VLC with WMA files.. Boom
  - But it’s obviously easier to find a crash than to figure out what caused it

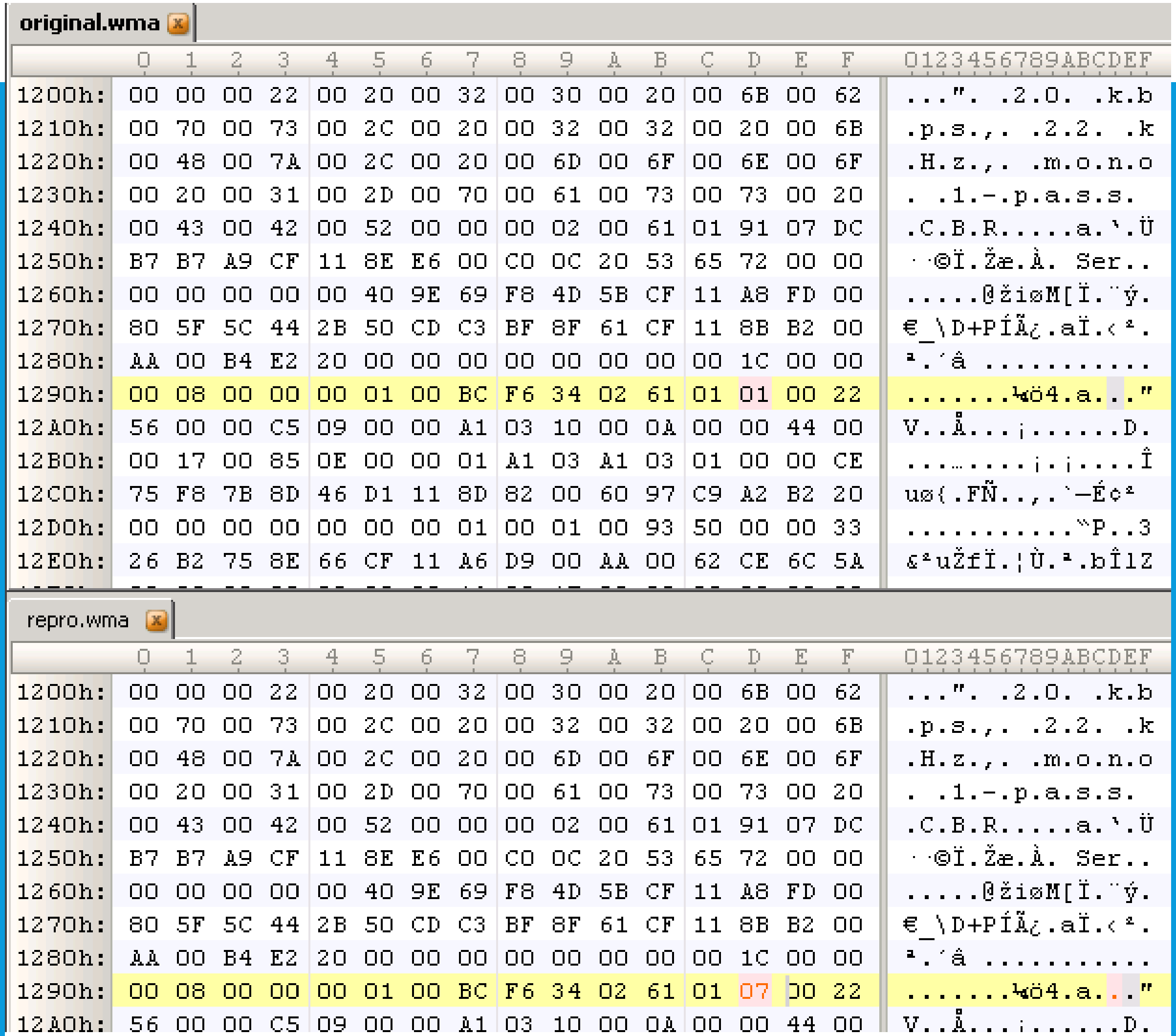
Reference: http://technet.microsoft.com/en-us/security/msvr/msvr12-017

CASE STUDY: LIBAVCODEC

- !Exploitable says a WriteAV at libavcodec_plugin.dll
  - Looks like this isn’t a bug in VLC, but in the included A/V codec

- Let’s diff to see what the fuzzer changed in the template to make our repro file!

CASE STUDY: LIBAVCODEC



original.wma

```
        0  1  2  3  4  5  6  7  8  9  A  B  C  D  E  F   0123456789ABCDEF
1200h: 00 00 00 22 00 20 00 32 00 30 00 20 00 6B 00 62  ...". .2.0. .k.b
1210h: 00 70 00 73 00 2C 00 20 00 32 00 32 00 20 00 6B  .p.s.,. .2.2. .k
1220h: 00 48 00 7A 00 2C 00 20 00 6D 00 6F 00 6E 00 6F  .H.z.,. .m.o.n.o
1230h: 00 20 00 31 00 2D 00 70 00 61 00 73 00 73 00 20  . .1.-.p.a.s.s. 
1240h: 00 43 00 42 00 52 00 00 00 02 00 61 01 91 07 DC  .C.B.R.....a.‘.Ü
1250h: B7 B7 A9 CF 11 8E E6 00 C0 0C 20 53 65 72 00 00  ··©Ï.Žæ.À. Ser..
1260h: 00 00 00 00 00 40 9E 69 F8 4D 5B CF 11 A8 FD 00  .....@žiøM[Ï.¨ý.
1270h: 80 5F 5C 44 2B 50 CD C3 BF 8F 61 CF 11 8B B2 00  €_\D+PÍÃ¿.aÏ.‹².
1280h: AA 00 B4 E2 20 00 00 00 00 00 00 00 00 1C 00 00  ª.´â ...........
1290h: 00 08 00 00 00 01 00 BC F6 34 02 61 01 01 00 22  .......¼ö4.a..."
12A0h: 56 00 00 C5 09 00 00 A1 03 10 00 0A 00 00 44 00  V..Å...¡......D.
12B0h: 00 17 00 85 0E 00 00 01 A1 03 A1 03 01 00 00 CE  ...…....¡.¡....Î
12C0h: 75 F8 7B 8D 46 D1 11 8D 82 00 60 97 C9 A2 B2 20  uø{.FÑ..‚.`—É¢² 
12D0h: 00 00 00 00 00 00 00 01 00 01 00 93 50 00 00 33  ...........“P..3
12E0h: 26 B2 75 8E 66 CF 11 A6 D9 00 AA 00 62 CE 6C 5A  &²uŽfÏ.¦Ù.ª.bÎlZ
```

repro.wma

```
        0  1  2  3  4  5  6  7  8  9  A  B  C  D  E  F   0123456789ABCDEF
1200h: 00 00 00 22 00 20 00 32 00 30 00 20 00 6B 00 62  ...". .2.0. .k.b
1210h: 00 70 00 73 00 2C 00 20 00 32 00 32 00 20 00 6B  .p.s.,. .2.2. .k
1220h: 00 48 00 7A 00 2C 00 20 00 6D 00 6F 00 6E 00 6F  .H.z.,. .m.o.n.o
1230h: 00 20 00 31 00 2D 00 70 00 61 00 73 00 73 00 20  . .1.-.p.a.s.s. 
1240h: 00 43 00 42 00 52 00 00 00 02 00 61 01 91 07 DC  .C.B.R.....a.‘.Ü
1250h: B7 B7 A9 CF 11 8E E6 00 C0 0C 20 53 65 72 00 00  ··©Ï.Žæ.À. Ser..
1260h: 00 00 00 00 00 40 9E 69 F8 4D 5B CF 11 A8 FD 00  .....@žiøM[Ï.¨ý.
1270h: 80 5F 5C 44 2B 50 CD C3 BF 8F 61 CF 11 8B B2 00  €_\D+PÍÃ¿.aÏ.‹².
1280h: AA 00 B4 E2 20 00 00 00 00 00 00 00 00 1C 00 00  ª.´â ...........
1290h: 00 08 00 00 00 01 00 BC F6 34 02 61 01 07 00 22  .......¼ö4.a..."
12A0h: 56 00 00 C5 09 00 00 A1 03 10 00 0A 00 00 44 00  V..Å...¡......D.
```

CASE STUDY: LIBAVCODEC

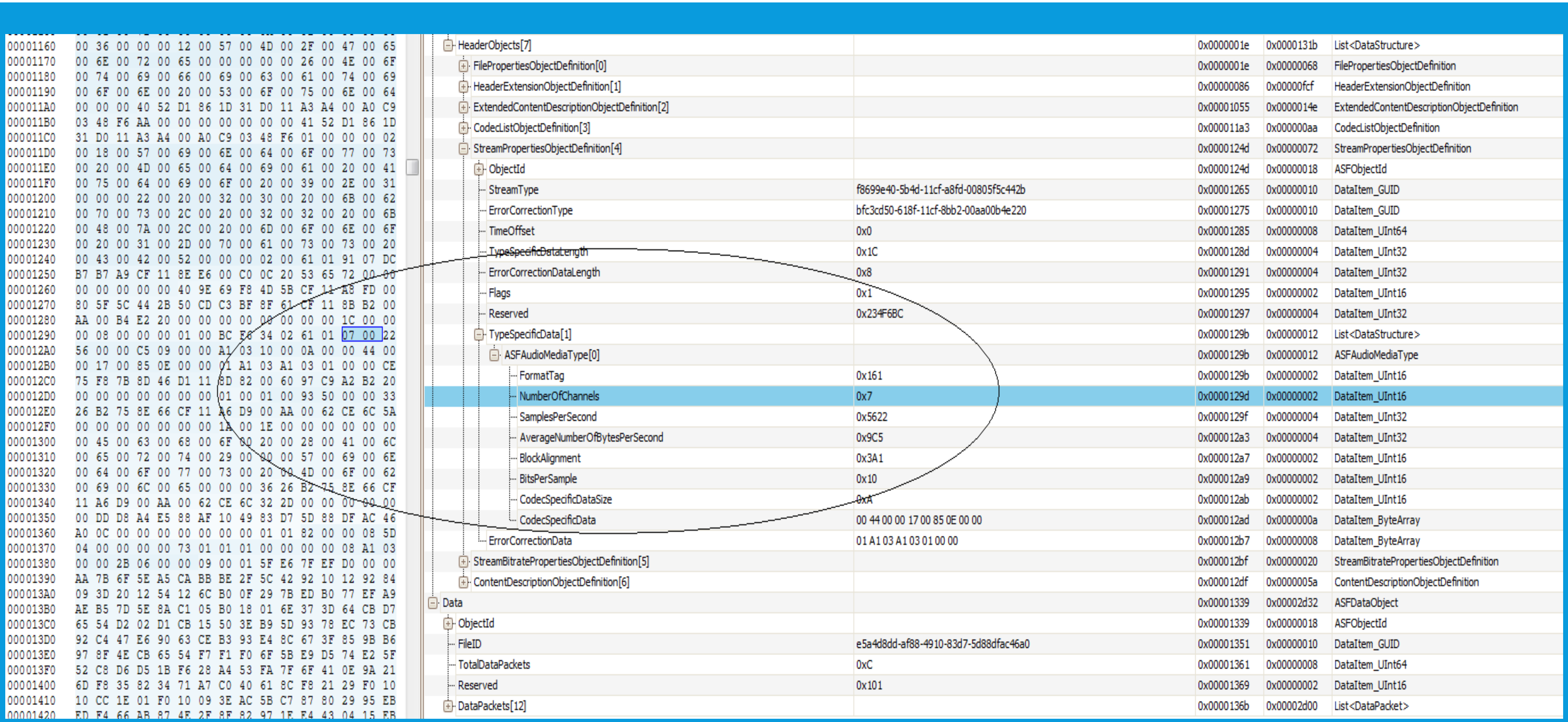
- We can see that the 0x0001 was changed to 0x0007
- But what is that word value anyways?
  - And how do I already know it's a word?

CASE STUDY: LIBAVCODEC

- Meet OffVis
  - "The Microsoft Office Visualization Tool (OffVis) allows IT professionals, security researchers and malware protection vendors to better understand the Microsoft Office binary file format in order to deconstruct .doc-, .xls- and .ppt-based targeted attacks"
  - Free public version available on the Microsoft download website
  - But it's not actually specifically for office documents. OffVis uses GUT templates, which is the same concept as 010 editor binary templates: describing file formats in order to parse and edit such files smarter.

Reference: http://www.microsoft.com/en-us/download/details.aspx?id=2096

CASE STUDY: LIBAVCODEC



```
00001160   00 36 00 00 00 12 00 57 00 4D 00 2F 00 47 00 65
00001170   00 6E 00 72 00 65 00 00 00 00 00 26 00 4E 00 6F
00001180   00 74 00 69 00 66 00 69 00 63 00 61 00 74 00 69
00001190   00 6F 00 6E 00 20 00 53 00 6F 00 75 00 6E 00 64
000011A0   00 00 00 40 52 D1 86 1D 31 D0 11 A3 A4 00 A0 C9
000011B0   03 48 F6 AA 00 00 00 00 00 00 00 41 52 D1 86 1D
000011C0   31 D0 11 A3 A4 00 A0 C9 03 48 F6 01 00 00 00 02
000011D0   00 18 00 57 00 69 00 6E 00 64 00 6F 00 77 00 73
000011E0   00 20 00 4D 00 65 00 64 00 69 00 61 00 20 00 41
000011F0   00 75 00 64 00 69 00 6F 00 20 00 39 00 2E 00 31
00001200   00 00 00 22 00 20 00 32 00 30 00 20 00 6B 00 62
00001210   00 70 00 73 00 2C 00 20 00 32 00 32 00 20 00 6B
00001220   00 48 00 7A 00 2C 00 20 00 6D 00 6F 00 6E 00 6F
00001230   00 20 00 31 00 2D 00 70 00 61 00 73 00 73 00 20
00001240   00 43 00 42 00 52 00 00 00 02 00 61 01 91 07 DC
00001250   B7 B7 A9 CF 11 8E E6 00 C0 0C 20 53 65 72 00 00
00001260   00 00 00 00 00 40 9E 69 F8 4D 5B CF 11 A8 FD 00
00001270   80 5F 5C 44 2B 50 CD C3 BF 8F 61 CF 11 8B B2 00
00001280   AA 00 B4 E2 20 00 00 00 00 00 00 00 00 1C 00 00
00001290   00 08 00 00 00 01 00 BC F6 34 02 61 01 07 00 22
000012A0   56 00 00 C5 09 00 00 A1 03 10 00 0A 00 00 44 00
000012B0   00 17 00 85 0E 00 00 01 A1 03 A1 03 01 00 00 CE
000012C0   75 F8 7B 8D 46 D1 11 8D 82 00 60 97 C9 A2 B2 20
000012D0   00 00 00 00 00 00 00 01 00 01 00 93 50 00 00 33
000012E0   26 B2 75 8E 66 CF 11 A6 D9 00 AA 00 62 CE 6C 5A
000012F0   00 00 00 00 00 00 00 1A 00 1E 00 00 00 00 00 00
00001300   00 45 00 63 00 68 00 6F 00 20 00 28 00 41 00 6C
00001310   00 65 00 72 00 74 00 29 00 00 00 57 00 69 00 6E
00001320   00 64 00 6F 00 77 00 73 00 20 00 4D 00 6F 00 62
00001330   00 69 00 6C 00 65 00 00 00 36 26 B2 75 8E 66 CF
00001340   11 A6 D9 00 AA 00 62 CE 6C 32 2D 00 00 00 00 00
00001350   00 DD D8 A4 E5 88 AF 10 49 83 D7 5D 88 DF AC 46
00001360   A0 0C 00 00 00 00 00 00 00 01 01 82 00 00 08 5D
00001370   04 00 00 00 00 73 01 01 01 00 00 00 00 08 A1 03
00001380   00 00 2B 06 00 00 09 00 01 5F E6 7F EF D0 00 00
00001390   AA 7B 6F 5E A5 CA BB BE 2F 5C 42 92 10 12 92 84
000013A0   09 3D 20 12 54 12 6C B0 0F 29 7B ED B0 77 EF A9
000013B0   AE B5 7D 5E 8A C1 05 B0 18 01 6E 37 3D 64 CB D7
000013C0   65 54 D2 02 D1 CB 15 50 3E B9 5D 93 78 EC 73 CB
000013D0   92 C4 47 E6 90 63 CE B3 93 E4 8C 67 3F 85 9B B6
000013E0   97 8F 4E CB 65 54 F7 F1 F0 6F 5B E9 D5 74 E2 5F
000013F0   52 C8 D6 D5 1B F6 28 A4 53 FA 7F 6F 41 0E 9A 21
00001400   6D F8 35 82 34 71 A7 C0 40 61 8C F8 21 29 F0 10
00001410   10 CC 1E 01 F0 10 09 3E AC 5B C7 87 80 29 95 EB
00001420   ED F4 66 AB 87 4E 2F 8F 82 97 1E E4 43 04 15 EB
```

HeaderObjects[7]		0x0000001e	0x0000131b	List<DataStructure>
FilePropertiesObjectDefinition[0]		0x0000001e	0x00000068	FilePropertiesObjectDefinition
HeaderExtensionObjectDefinition[1]		0x00000086	0x00000fcf	HeaderExtensionObjectDefinition
ExtendedContentDescriptionObjectDefinition[2]		0x00001055	0x0000014e	ExtendedContentDescriptionObjectDefinition
CodecListObjectDefinition[3]		0x000011a3	0x000000aa	CodecListObjectDefinition
StreamPropertiesObjectDefinition[4]		0x0000124d	0x00000072	StreamPropertiesObjectDefinition
ObjectId		0x0000124d	0x00000018	ASFObjectId
StreamType	f8699e40-5b4d-11cf-a8fd-00805f5c442b	0x00001265	0x00000010	DataItem_GUID
ErrorCorrectionType	bfc3cd50-618f-11cf-8bb2-00aa00b4e220	0x00001275	0x00000010	DataItem_GUID
TimeOffset	0x0	0x00001285	0x00000008	DataItem_UInt64
TypeSpecificDataLength	0x1C	0x0000128d	0x00000004	DataItem_UInt32
ErrorCorrectionDataLength	0x8	0x00001291	0x00000004	DataItem_UInt32
Flags	0x1	0x00001295	0x00000002	DataItem_UInt16
Reserved	0x234F6BC	0x00001297	0x00000004	DataItem_UInt32
TypeSpecificData[1]		0x0000129b	0x00000012	List<DataStructure>
ASFAudioMediaType[0]		0x0000129b	0x00000012	ASFAudioMediaType
FormatTag	0x161	0x0000129b	0x00000002	DataItem_UInt16
NumberOfChannels	0x7	0x0000129d	0x00000002	DataItem_UInt16
SamplesPerSecond	0x5622	0x0000129f	0x00000004	DataItem_UInt32
AverageNumberOfBytesPerSecond	0x9C5	0x000012a3	0x00000004	DataItem_UInt32
BlockAlignment	0x3A1	0x000012a7	0x00000002	DataItem_UInt16
BitsPerSample	0x10	0x000012a9	0x00000002	DataItem_UInt16
CodecSpecificDataSize	0xA	0x000012ab	0x00000002	DataItem_UInt16
CodecSpecificData	00 44 00 00 17 00 85 0E 00 00	0x000012ad	0x0000000a	DataItem_ByteArray
ErrorCorrectionData	01 A1 03 A1 03 01 00 00	0x000012b7	0x00000008	DataItem_ByteArray
StreamBitratePropertiesObjectDefinition[5]		0x000012bf	0x00000020	StreamBitratePropertiesObjectDefinition
ContentDescriptionObjectDefinition[6]		0x000012df	0x0000005a	ContentDescriptionObjectDefinition
Data		0x00001339	0x00002d32	ASFDataObject
ObjectId		0x00001339	0x00000018	ASFObjectId
FileID	e5a4d8dd-af88-4910-83d7-5d88dfac46a0	0x00001351	0x00000010	DataItem_GUID
TotalDataPackets	0xC	0x00001361	0x00000008	DataItem_UInt64
Reserved	0x101	0x00001369	0x00000002	DataItem_UInt16
DataPackets[12]		0x0000136b	0x00002d00	List<DataPacket>

CASE STUDY: LIBAVCODEC

- So we know a few more things now!
  - ASF is the container format for WMA files
  - A quick search for "Number of Channels" in the ASF specification tells us
    - It's a 16-bit value
    - It's a member of the WAVEFORMATEX structure
    - It's the "number of audio channels" for this content
  - Manual testing shows that changing the value from 0x0003 - 0x0008 causes a crash
    - Also noteworthy, changing it to 0x0009 results in VLC displaying an error dialog about how VLC does not support the WMA2 file format

Reference: http://msdn.microsoft.com/en-us/library/bb643323.aspx

CASE STUDY: LIBAVCODEC

- Now take a look at the couple of instructions before the crash
  - pop ebx
  - call dword ptr[ebx+30h]
- Anyone else smiling? ☺
- For those not immediately enlightened, this is very promising for exploitation
  - As long as we have some kind of influence or control over the ebx register
  - And there's a pop before the call.. well, the stack is our friend

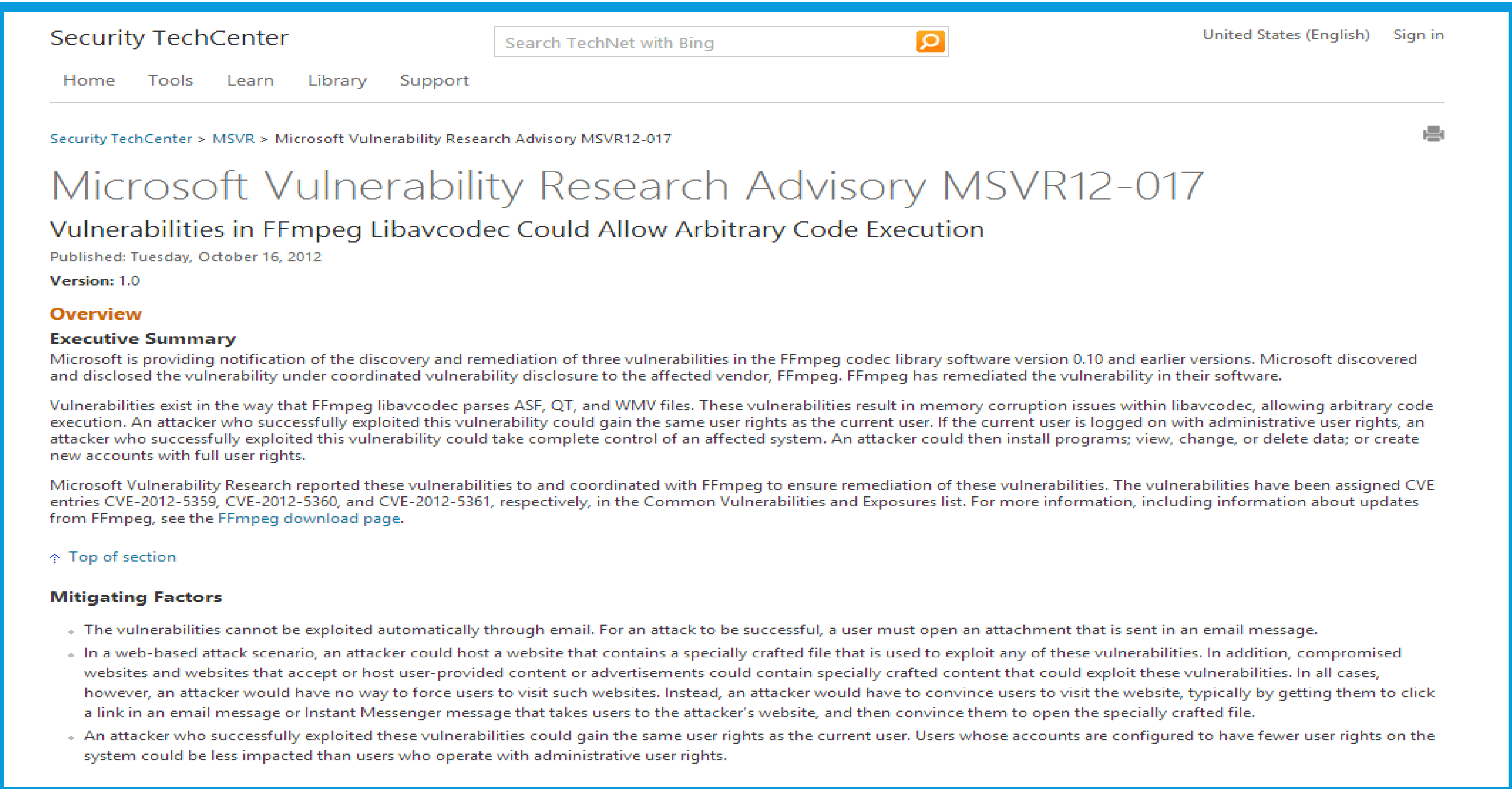
CASE STUDY: LIBAVCODEC

- We've got our original and repro files, quick write-up and ready to share with msvr@microsoft.com
- They packaged up the deliverables and sent them off to the vendor
  - Handled coordination
  - Status updates
  - Questions from the vendor

CASE STUDY: LIBAVCODEC

- The vulnerability was patched in May, 2012 and the advisory was released a few months later

CASE STUDY: LIBAVCODEC



Security TechCenter > MSVR > Microsoft Vulnerability Research Advisory MSVR12-017

Microsoft Vulnerability Research Advisory MSVR12-017

Vulnerabilities in FFmpeg Libavcodec Could Allow Arbitrary Code Execution

Published: Tuesday, October 16, 2012

Version: 1.0

Overview

Executive Summary

Microsoft is providing notification of the discovery and remediation of three vulnerabilities in the FFmpeg codec library software version 0.10 and earlier versions. Microsoft discovered and disclosed the vulnerability under coordinated vulnerability disclosure to the affected vendor, FFmpeg. FFmpeg has remediated the vulnerability in their software.

Vulnerabilities exist in the way that FFmpeg libavcodec parses ASF, QT, and WMV files. These vulnerabilities result in memory corruption issues within libavcodec, allowing arbitrary code execution. An attacker who successfully exploited this vulnerability could gain the same user rights as the current user. If the current user is logged on with administrative user rights, an attacker who successfully exploited this vulnerability could take complete control of an affected system. An attacker could then install programs; view, change, or delete data; or create new accounts with full user rights.

Microsoft Vulnerability Research reported these vulnerabilities to and coordinated with FFmpeg to ensure remediation of these vulnerabilities. The vulnerabilities have been assigned CVE entries CVE-2012-5359, CVE-2012-5360, and CVE-2012-5361, respectively, in the Common Vulnerabilities and Exposures list. For more information, including information about updates from FFmpeg, see the FFmpeg download page.

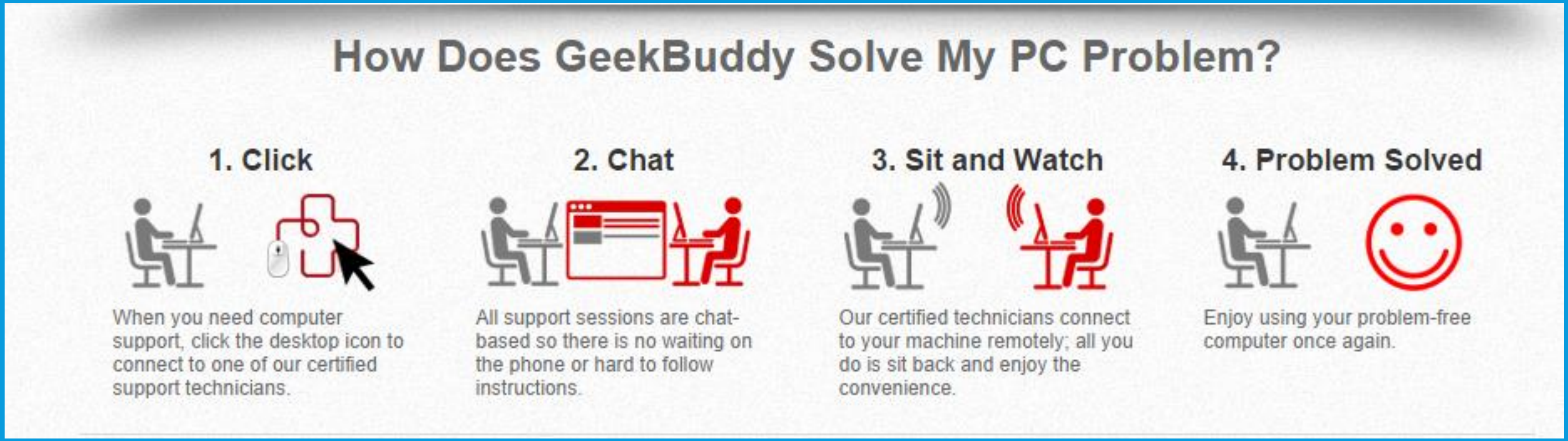
Top of section

Mitigating Factors

- The vulnerabilities cannot be exploited automatically through email. For an attack to be successful, a user must open an attachment that is sent in an email message.
- In a web-based attack scenario, an attacker could host a website that contains a specially crafted file that is used to exploit any of these vulnerabilities. In addition, compromised websites and websites that accept or host user-provided content or advertisements could contain specially crafted content that could exploit these vulnerabilities. In all cases, however, an attacker would have no way to force users to visit such websites. Instead, an attacker would have to convince users to visit the website, typically by getting them to click a link in an email message or Instant Messenger message that takes users to the attacker's website, and then convince them to open the specially crafted file.
- An attacker who successfully exploited these vulnerabilities could gain the same user rights as the current user. Users whose accounts are configured to have fewer user rights on the system could be less impacted than users who operate with administrative user rights.

CASE STUDY: COMODO GEEKBUDDY

- CVE-2014-7872
  - Comodo GeekBuddy Privilege Escalation
- What is GeekBuddy and how does it work?



How Does GeekBuddy Solve My PC Problem?

1. Click

When you need computer support, click the desktop icon to connect to one of our certified support technicians.

2. Chat

All support sessions are chat-based so there is no waiting on the phone or hard to follow instructions.

3. Sit and Watch

Our certified technicians connect to your machine remotely; all you do is sit back and enjoy the convenience.

4. Problem Solved

Enjoy using your problem-free computer once again.

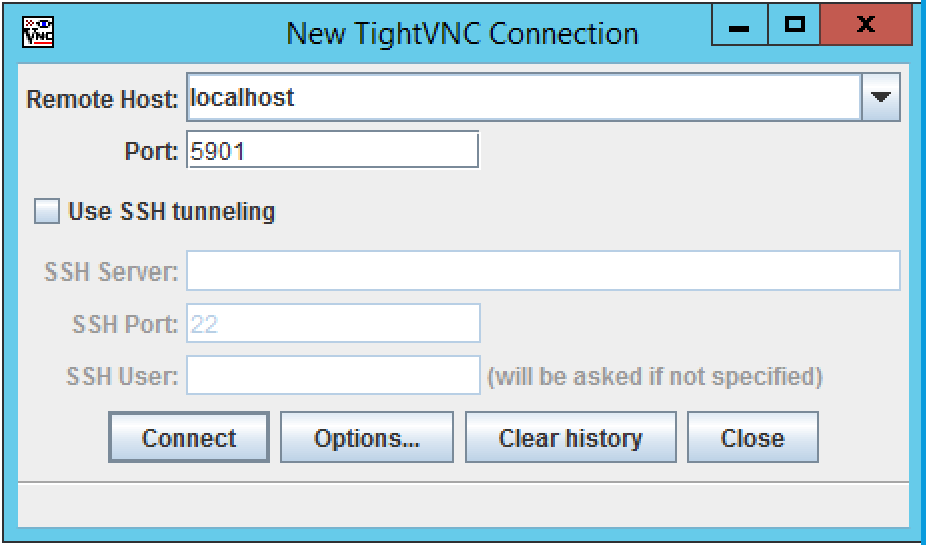
CASE STUDY: COMODO GEEKBUDDY

- Noticed GeekBuddyRSP.exe was listening on two familiar ports
  - 5800, 5901 (VNC)

- VNC server to tunnel technical support remoting makes sense

CASE STUDY: COMODO GEEKBUDDY

- Let's try to connect using a VNC client and see what happens



Remote Host: localhost

Port: 5901

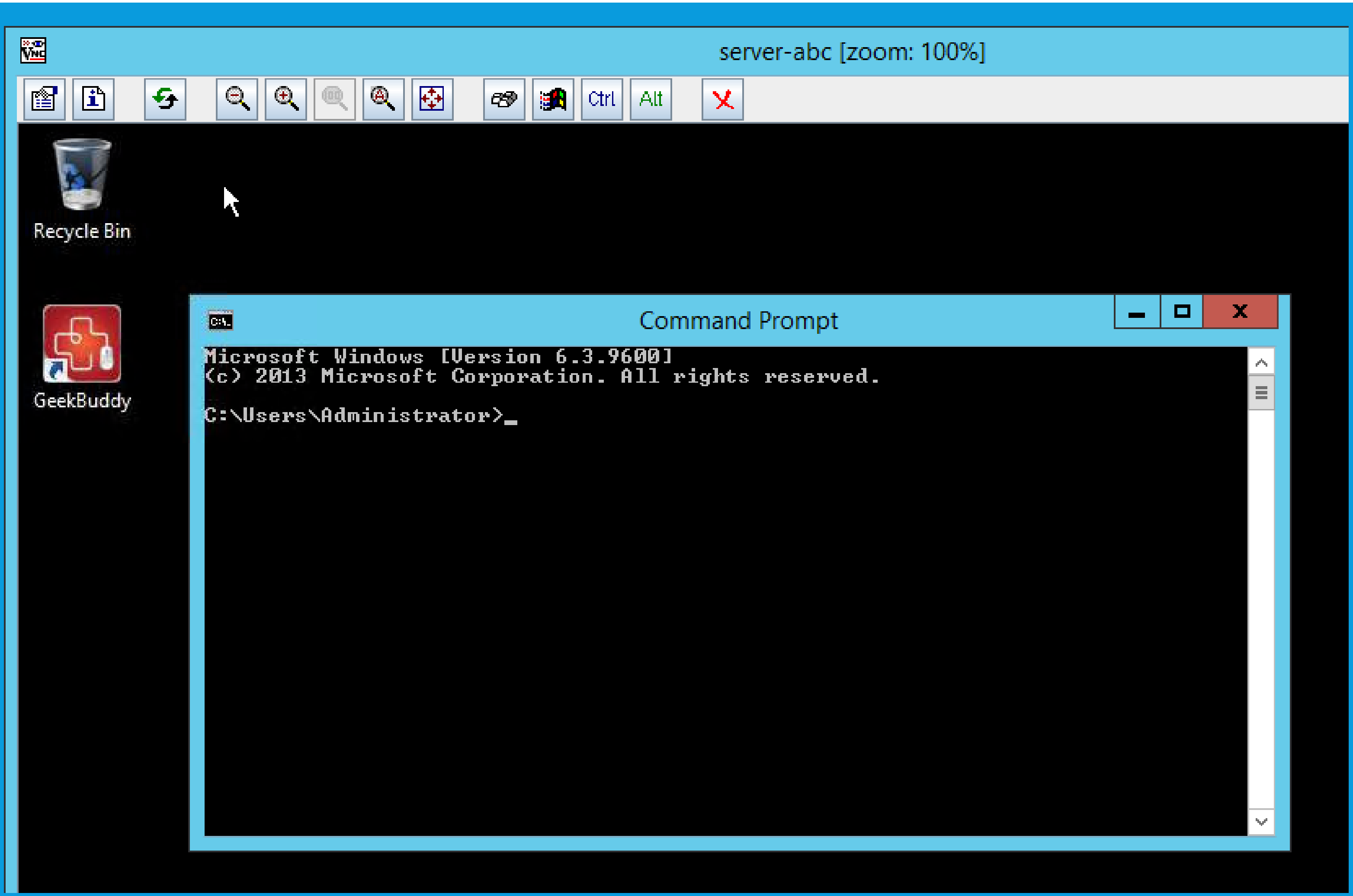
☐ Use SSH tunneling

SSH Server:

SSH Port: 22

SSH User: (will be asked if not specified)

CASE STUDY: COMODO GEEKBUDDY



```
Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.

C:\Users\Administrator>_
```

CASE STUDY: COMODO GEEKBUDDY

- The attack goes as follows
  - Admin logs in
  - User (or guest) logs in and uses a VNC client to connect to localhost
  - User assumes Administrator's VNC session via **no server password set**

  - Couple significant caveats
    - OS must support more than one simultaneous login, eg. Windows Server
    - GeekBuddy is known to be bundled with the following products
      - Comodo Anti-Virus, Comodo Firewall, Comodo Internet Security
      - But they only install on Windows Client
    - Comodo might have bundled GeekBuddy in some enterprise packages

CASE STUDY: COMODO GEEKBUDDY

- What other vectors of exploitation can you think of?
- Client-side CSRF-like attack
  - Host a modified Java VNC Client on a webserver
  - GeekBuddy target browses to webpage with embedded VNC client
  - VNC client connects to localhost and does interesting things with the target's session
- Comodo released a fixed version October, 2014

CASE STUDY: VMWARE

- MSVR13-002
  - Vulnerabilities in VMware OVF Tool Could Allow Arbitrary Code Execution
- Step 1: What file types does VMware handle?
  - VMX
  - VMDK
  - OVF
  - …more

Reference: http://technet.microsoft.com/en-us/security/msvr/msvr13-002

CASE STUDY: VMWARE

- Step 2: What is OVF?
  - Open Virtual Machine Format
  - "an open, secure, portable, efficient and extensible format for the packing and distribution of (collections of) virtual machines"

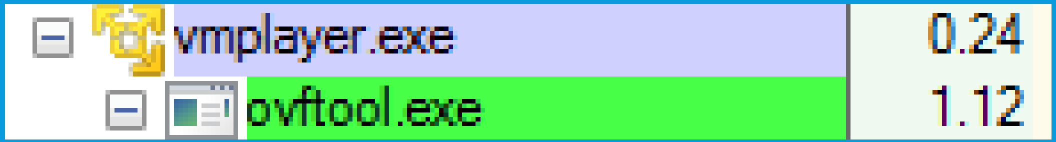
Reference: http://www.vmware.com/pdf/ovf_whitepaper_specification.pdf

CASE STUDY: VMWARE

- TL;DR– It’s a xml-based file format for describing virtual machine data
  - And since XML implies describing and consuming untrusted data.. probably a worthy target

CASE STUDY: VMWARE

- Step 3: How does VMware load OVF files?

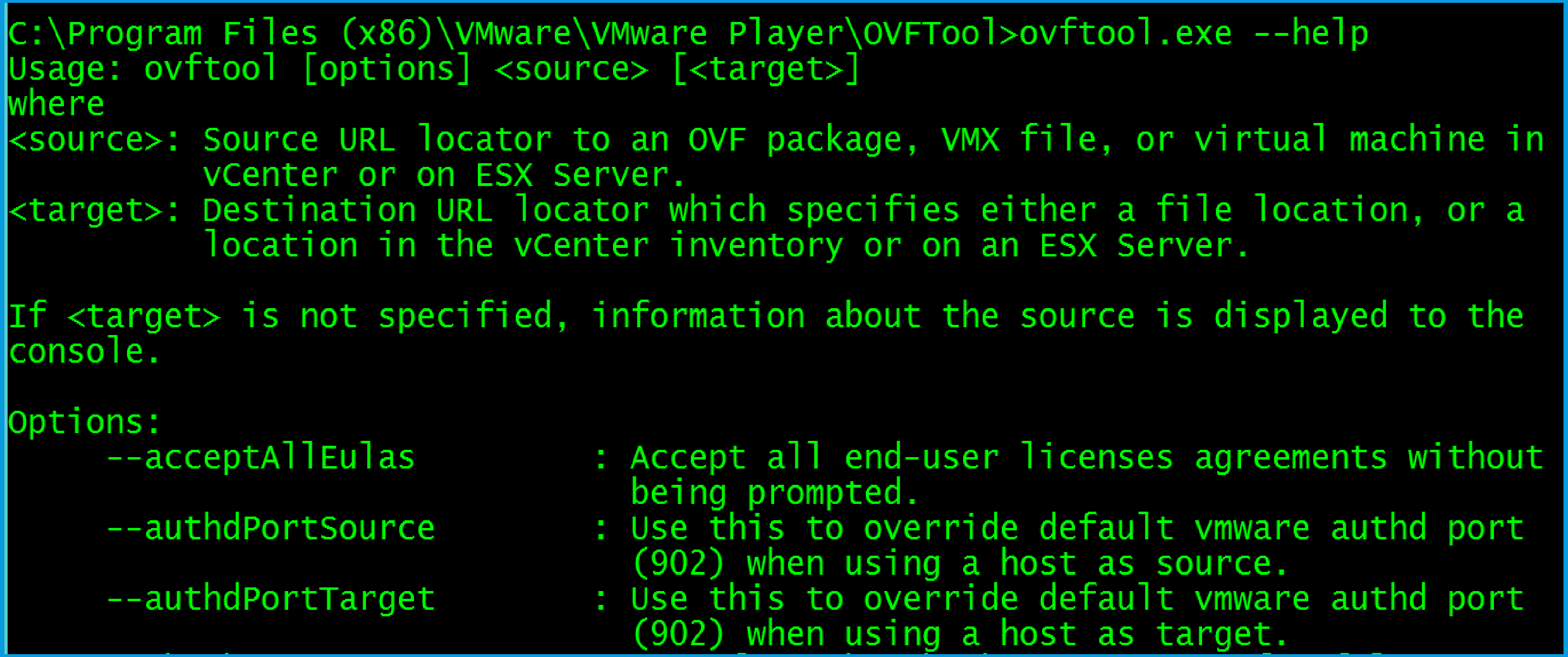


```
vmplayer.exe      0.24
  ovftool.exe     1.12
```

- Upon loading a OVF file, it executes ovftool.exe
  - Nearly the same as having the OVF parsing code in VMware player

CASE STUDY: VMWARE

Step 4: What is OVFTool?



```
C:\Program Files (x86)\VMware\VMware Player\OVFTool>ovftool.exe --help
Usage: ovftool [options] <source> [<target>]
where
<source>: Source URL locator to an OVF package, VMX file, or virtual machine in
          vCenter or on ESX Server.
<target>: Destination URL locator which specifies either a file location, or a
          location in the vCenter inventory or on an ESX Server.

If <target> is not specified, information about the source is displayed to the
console.

Options:
     --acceptAllEulas          : Accept all end-user licenses agreements without
                                 being prompted.
     --authdPortSource         : Use this to override default vmware authd port
                                 (902) when using a host as source.
     --authdPortTarget         : Use this to override default vmware authd port
                                 (902) when using a host as target.
```

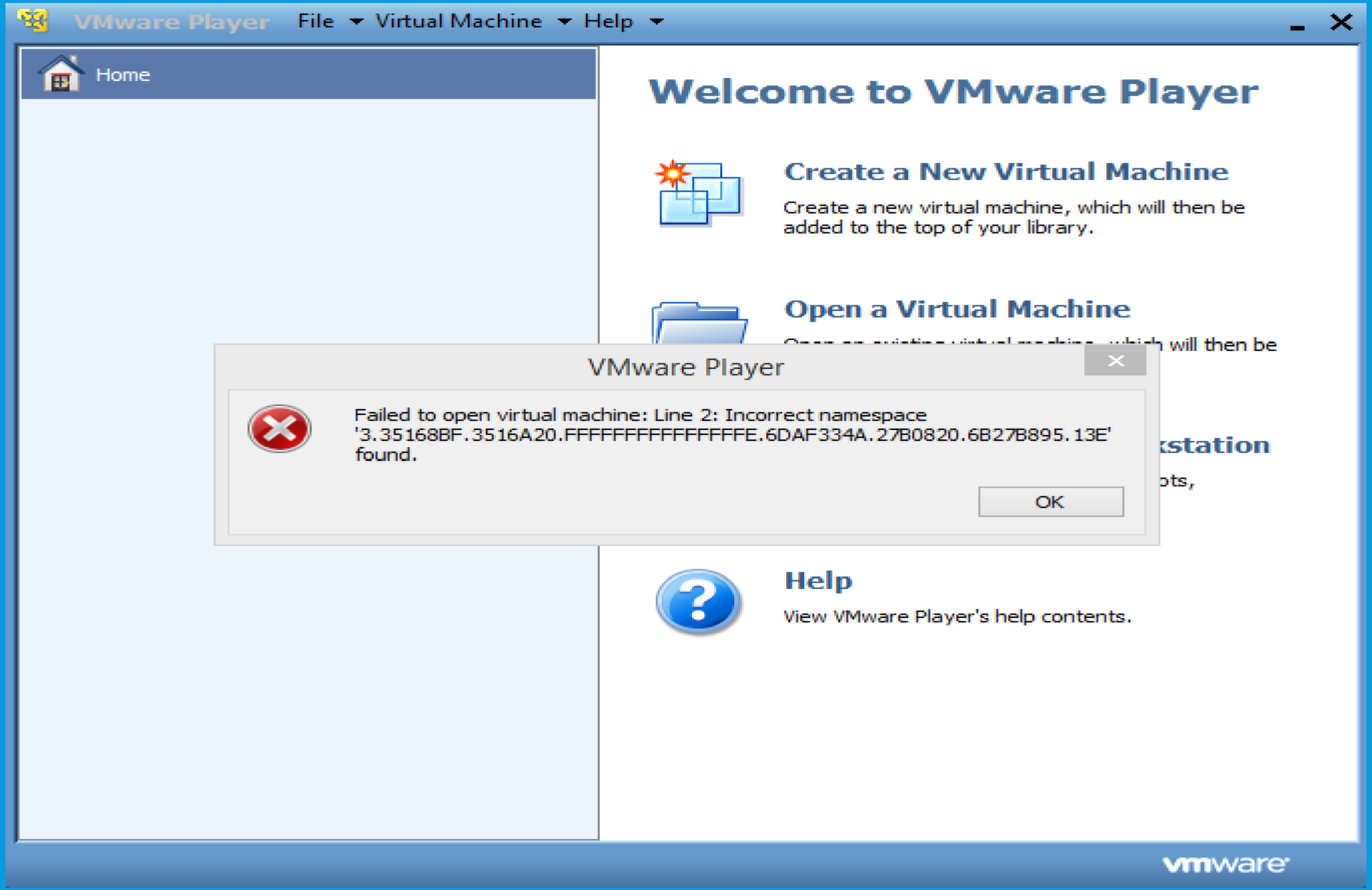
CASE STUDY: VMWARE

Step 5: Find a interesting crash or other unexpected behavior

```
<?xml version="1.0" encoding="utf-8"?>
<ovf:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:ovf="%p.%p.%p.%p.%p.%p.%p.%p"
xmlns:vssd="http://schemas.dmtf.org/wbem/wscim/1/cim-schema/2/CIM_VirtualSystemSettingData"
…
</ovf:Envelope>
```

CASE STUDY: VMWARE

And when we load the OVF file in VMware..



Welcome to VMware Player

Create a New Virtual Machine

Create a new virtual machine, which will then be added to the top of your library.

Open a Virtual Machine

will then be

station

ots,

Help

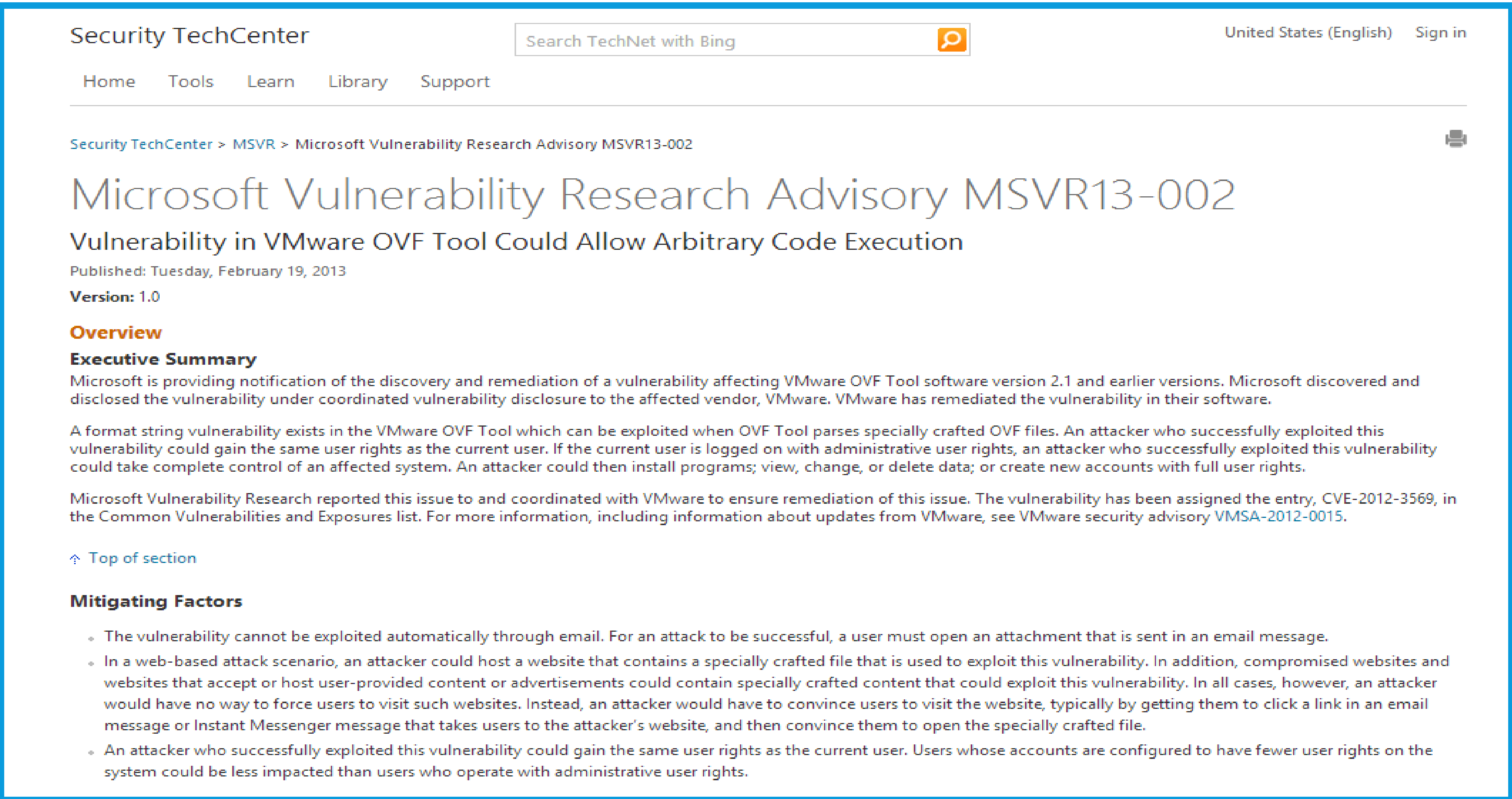
View VMware Player's help contents.

VMware Player

Failed to open virtual machine: Line 2: Incorrect namespace '3.35168BF.3516A20.FFFFFFFFFFFFFFFE.6DAF334A.27B0820.6B27B895.13E' found.

OK

CASE STUDY: VMWARE



Microsoft Vulnerability Research Advisory MSVR13-002

Vulnerability in VMware OVF Tool Could Allow Arbitrary Code Execution

Published: Tuesday, February 19, 2013

Version: 1.0

Overview

Executive Summary

Microsoft is providing notification of the discovery and remediation of a vulnerability affecting VMware OVF Tool software version 2.1 and earlier versions. Microsoft discovered and disclosed the vulnerability under coordinated vulnerability disclosure to the affected vendor, VMware. VMware has remediated the vulnerability in their software.

A format string vulnerability exists in the VMware OVF Tool which can be exploited when OVF Tool parses specially crafted OVF files. An attacker who successfully exploited this vulnerability could gain the same user rights as the current user. If the current user is logged on with administrative user rights, an attacker who successfully exploited this vulnerability could take complete control of an affected system. An attacker could then install programs; view, change, or delete data; or create new accounts with full user rights.

Microsoft Vulnerability Research reported this issue to and coordinated with VMware to ensure remediation of this issue. The vulnerability has been assigned the entry, CVE-2012-3569, in the Common Vulnerabilities and Exposures list. For more information, including information about updates from VMware, see VMware security advisory VMSA-2012-0015.

Top of section

Mitigating Factors

- The vulnerability cannot be exploited automatically through email. For an attack to be successful, a user must open an attachment that is sent in an email message.
- In a web-based attack scenario, an attacker could host a website that contains a specially crafted file that is used to exploit this vulnerability. In addition, compromised websites and websites that accept or host user-provided content or advertisements could contain specially crafted content that could exploit this vulnerability. In all cases, however, an attacker would have no way to force users to visit such websites. Instead, an attacker would have to convince users to visit the website, typically by getting them to click a link in an email message or Instant Messenger message that takes users to the attacker's website, and then convince them to open the specially crafted file.
- An attacker who successfully exploited this vulnerability could gain the same user rights as the current user. Users whose accounts are configured to have fewer user rights on the system could be less impacted than users who operate with administrative user rights.

CASE STUDY: BLACKBERRY PTG

- Submitted as, “Blackberry Print To Go Auth Bypass”
  - But what can we gain from this bug?

- What is Blackberry PTG?
  - Allows you to “print” documents from your computer to your BlackBerry Playbook tablet
  - E.g. Install the software on your PC and you can send anything you can print as a PDF to your Playbook

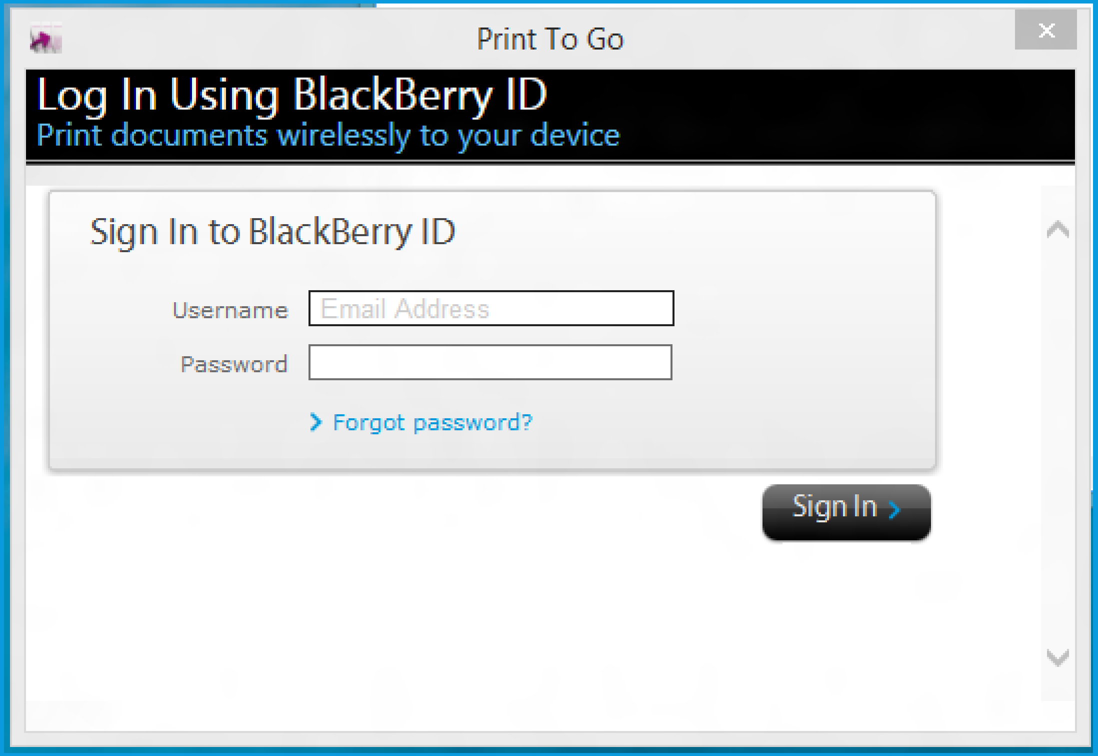
CASE STUDY: BLACKBERRY PTG

CASE STUDY: BLACKBERRY PTG

- In order to send documents to the Playbook, the user must do the following
  - Log into the service using your BlackBerry ID (user/pass)
  - Encrypt the documents using a password generated from the PTG app on the Playbook
  - Find the device using the it’s PIN

- We can bypass this locally
  - Therefore we won’t need to login to Blackberry to perhaps “print” documents to a device

CASE STUDY: BLACKBERRY PTG



Print To Go

Log In Using BlackBerry ID

Print documents wirelessly to your device

Sign In to BlackBerry ID

Username Email Address

Password

CASE STUDY: BLACKBERRY PTG

- There’s something listening on port 1234.. interesting

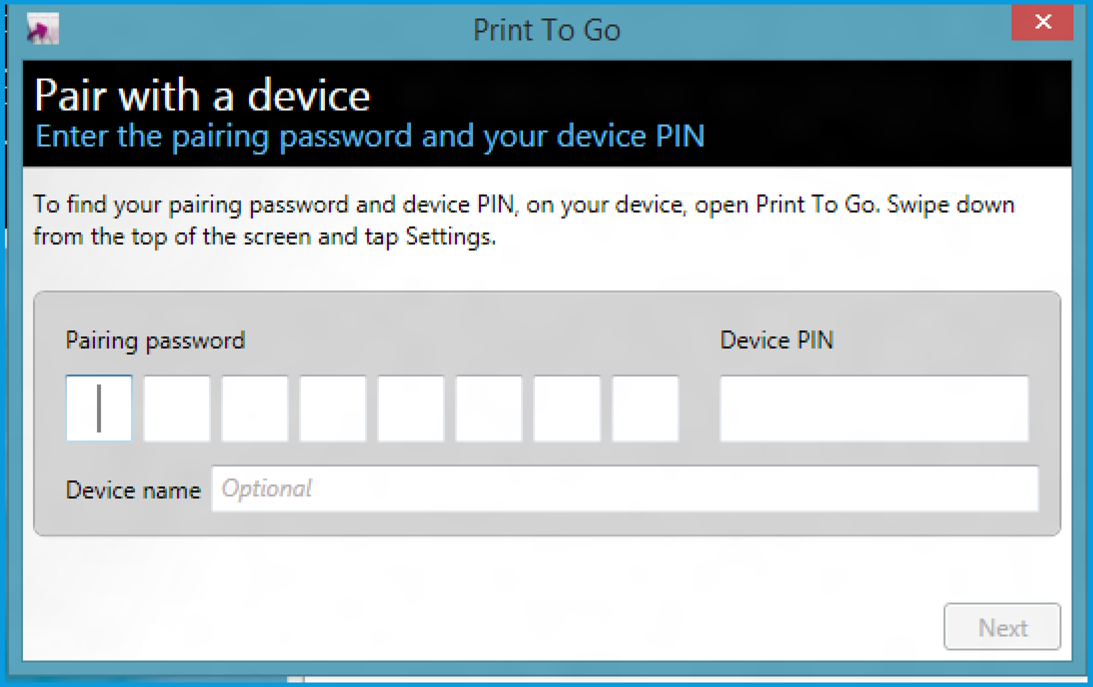
- With the BB login dialog open, start a web browser and simply point it to this URL
  - http://localhost:1234/myserverlet/

- The login dialog will immediately continue to the next page
  - Therefore bypassing authentication

CASE STUDY: BLACKBERRY PTG

- Theory
  - The login procedure checks if it *receives* data on listening port 1234, not the data's validity (at least well enough)

CASE STUDY: BLACKBERRY PTG



Pair with a device

Enter the pairing password and your device PIN

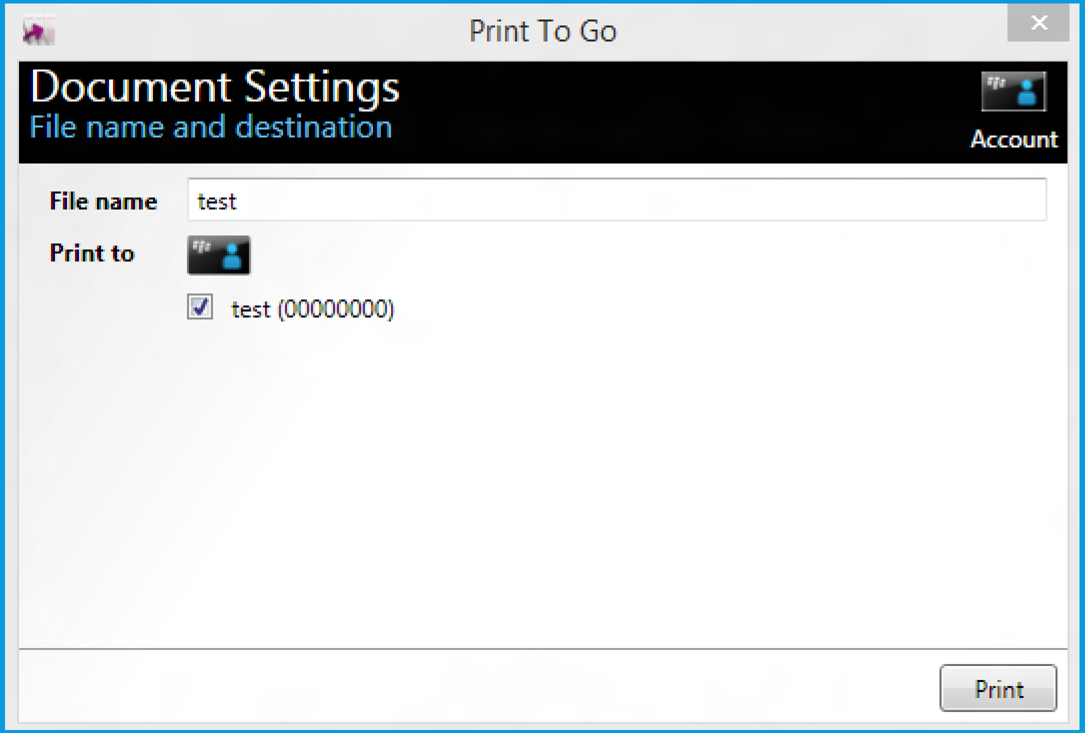
To find your pairing password and device PIN, on your device, open Print To Go. Swipe down from the top of the screen and tap Settings.

Pairing password

Device PIN

Device name Optional

CASE STUDY: BLACKBERRY PTG



Print To Go

Document Settings

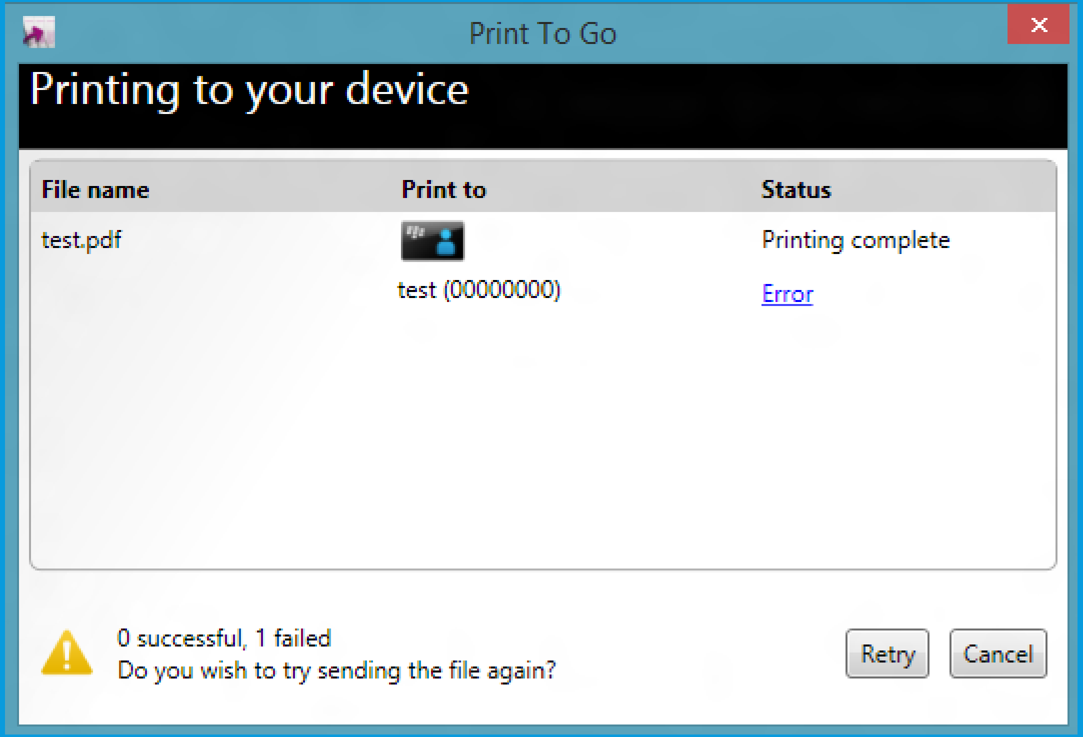
File name and destination

File name: test

Print to

☑ test (00000000)

CASE STUDY: BLACKBERRY PTG



Printing to your device

File name	Print to	Status
test.pdf	test (00000000)	Printing complete Error

0 successful, 1 failed
Do you wish to try sending the file again?

CASE STUDY: BLACKBERRY PTG

- So what could one gain from bypassing this login page?
  - There wasn't a Playbook tablet to completely test the exploit scenarios
  - We handed the report to BlackBerry security with our ideas so they could test internally
- BB concluded that while this was undesirable behavior, it wasn't a security issue
  - "Printing does not succeed as the Connector does not have the BlackBerry ID account info and token needed for printing"
- Without a Playbook on hand, it was tough to test this remaining step
  - We didn't know if it would succeed or not with a real device connected
  - Better to submit anyways so they could confirm with us

LESSONS LEARNED

- Vendors range greatly in their capacity
  - Which is not necessarily correlated with size
    - Some small development teams are very responsive, others are not
    - Some big companies have effective and established procedures, others mire you in bureaucracy

LESSONS LEARNED

- Setting limits is important
  - Pen-testing the web and dumped hundreds of bugs on us for most for relatively unimportant sites doesn’t scale too well
  - Finders may report low-severity bugs that they think are very serious
- Employees like this program!

WHY YOU SHOULD RUN YOUR OWN MSVR

- Give employees a standard, end-to-end process for getting security bugs fixed
- Inter-company bug reporting can be more coordinated and efficient
- Relatively cheap to run, with high ROI
- Boost employee morale
- Secure the ecosystem, as your product likely depends on *something*
  - Eg. HackerOne bug bounty program has a bounty for “The Internet”

WHAT WE'D LIKE TO SEE WHEN REPORTING VULNERABILITIES

- Clearly identified point of contact
- Public encryption key (PGP or S/MIME)
- Direct line to a real person who understands security
  - Don't turn us away because we don't have a support contract!

WHAT WE'D LIKE TO SEE WHEN REPORTING VULNERABILITIES

- Clear communication
  - Acknowledgment receipt of the initial email
  - Repro, including affected platforms
  - Update release dates, including any delays
  - How we will be credited (ask us for our preference!)
  - Closure
- Variant investigation
- Relatively prompt fixes

QUESTIONS?

CONTACT

- msvr@microsoft.com

This document is provided "as-is." Information and views expressed in this document, including URL and other Internet Web site references, may change without notice. You bear the risk of using it. Some examples are for illustration only and are fictitious. No real association is intended or inferred. This document does not provide you with any legal rights to any intellectual property owned by Microsoft. You may copy and use this document for your internal, reference purposes.