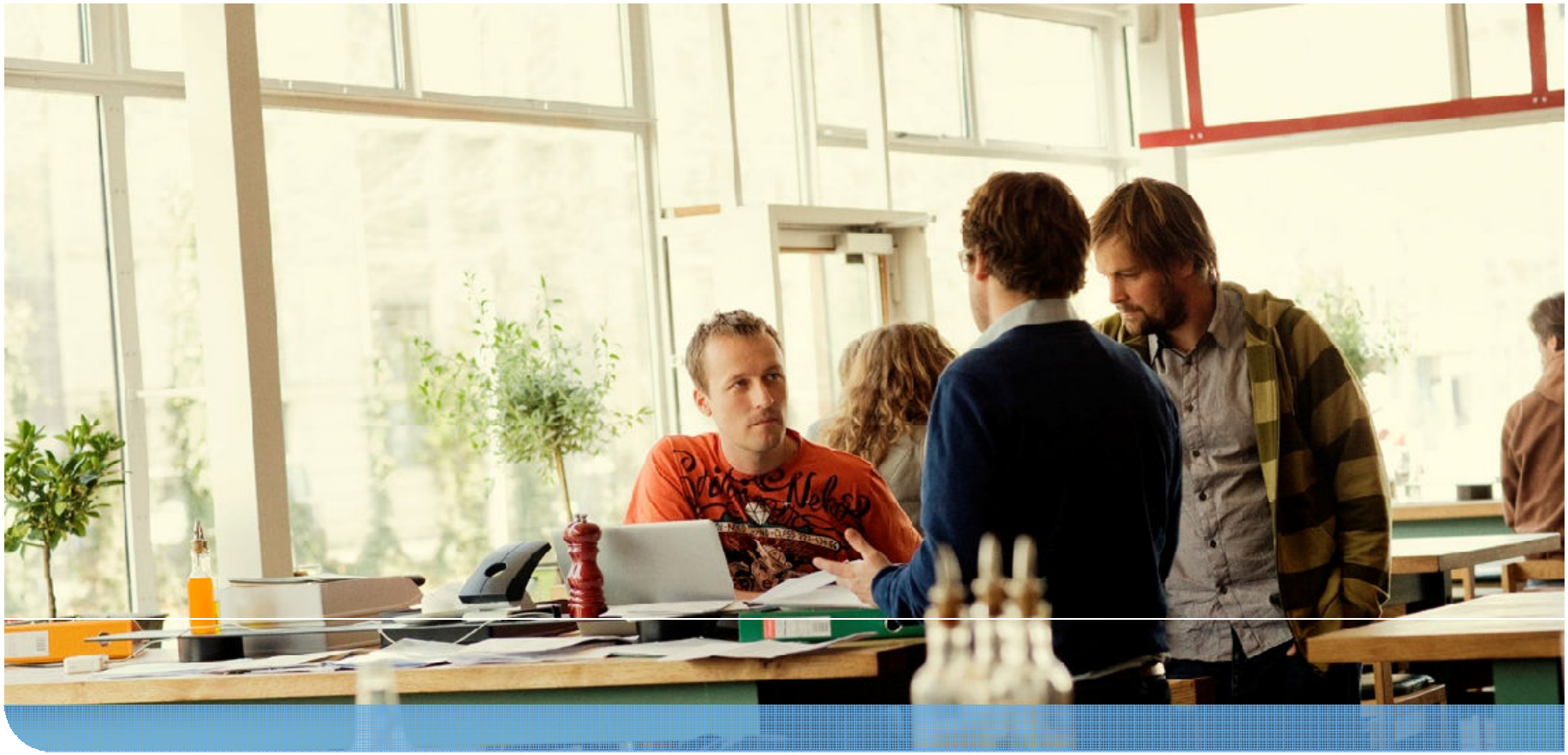


It's Signed, therefore it's Clean, right?

Jarno Niemelä
CARO 2010

F-Secure

Authenticode

- Code signing infrastructure for Microsoft Windows OS
  - Introduced in Windows NT
  - Actively required since Windows Vista
- Authenticode ensures code authenticity and integrity
  - A guarantee of software origin and that it has not been tampered
  - Common assumption is that if code is signed it can be trusted
- Microsoft has been pushing developers to sign their code
  - If developers want to get Windows logo code has to be signed
  - Which means that many developers treat this as nuisance

Authenticode from AV point of view

- Since Authenticode is crypto, techies tend to trust it
  - And this also includes people working in AV companies
- Thus, AV companies tend to use Authenticode to avoid FAs
  - Valid signature is strong indication of FA
  - Automation systems usually avoid signed files
    - Either intentionally or as result of bias given by learning set
- However, Authenticode is also useful for detection purposes
  - Cert that is used only in malware/PUP gives 100% detection rate
  - Thus just any cert won't do for malware, it has to be one that makes AV to scratch it's head for a while

What's This Mean For Malware Authors?

- Modern IE and Windows versions require signed binaries
  - Installing drivers without warning on 32-bit Windows Vista and 7
  - To be able to install driver at all in 64-bit versions for Vista and 7
  - Installing ActiveX components without warning
  - Or to be able to install them at all with tighter configurations
- Signed code is considered to be more trustworthy
  - Users are more likely to install software without scary warnings
  - AV companies are vary of files with legitimate looking signature
- Thus having valid signature that is associated with clean activity can mean slower reaction time from security vendors

The Number Of Signed Unwanted Files

- In F-Secures sample collection we have following files that are detected by us or at least two major vendors
- Potentially unwanted programs
  - Dialers, toolbars, adware, spyware and other unwanted programs
  - 384935 files
- Malware
  - Files that no vendor detects as potentially unwanted
  - 23817 files
- In this research we focus on malware

Ways Of Abusing Authenticode

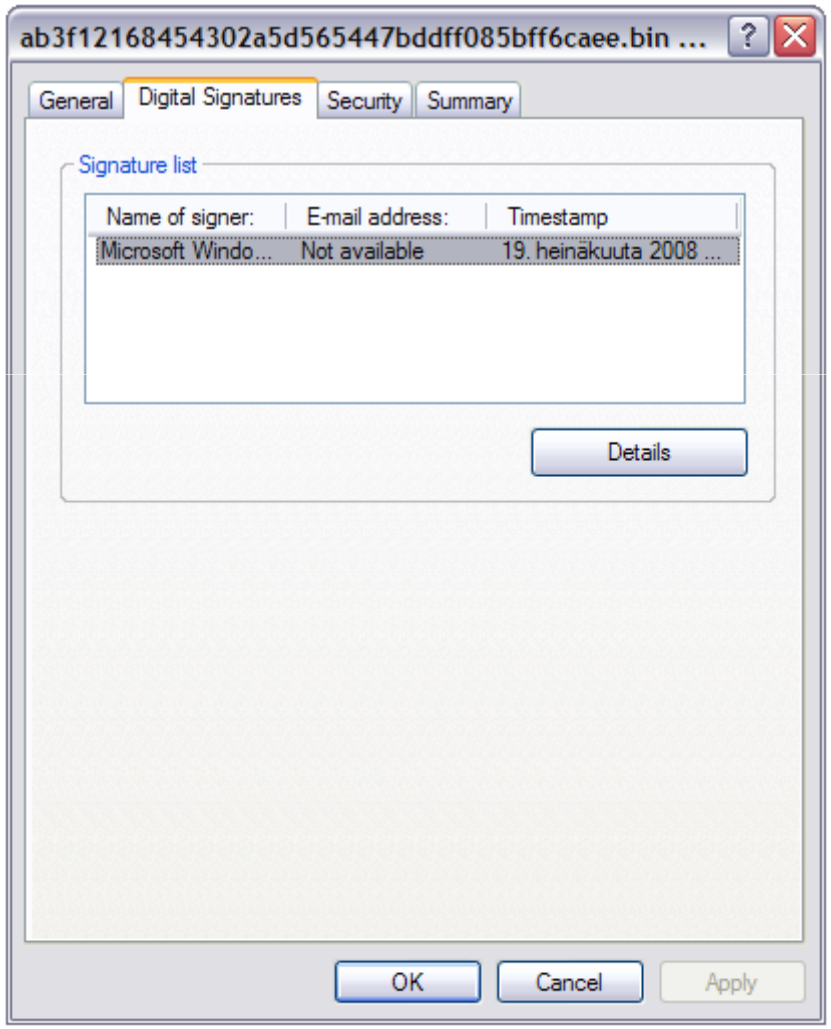
- Copying Certificate information from clean files
- Selfsigned certs with fake name
- MD5 forgery
- Get certified and be evil
- Get certificate with misleading name
- Find someone to sign your stuff for you
- Steal a certificate
- Infect developers system and get signed with software release

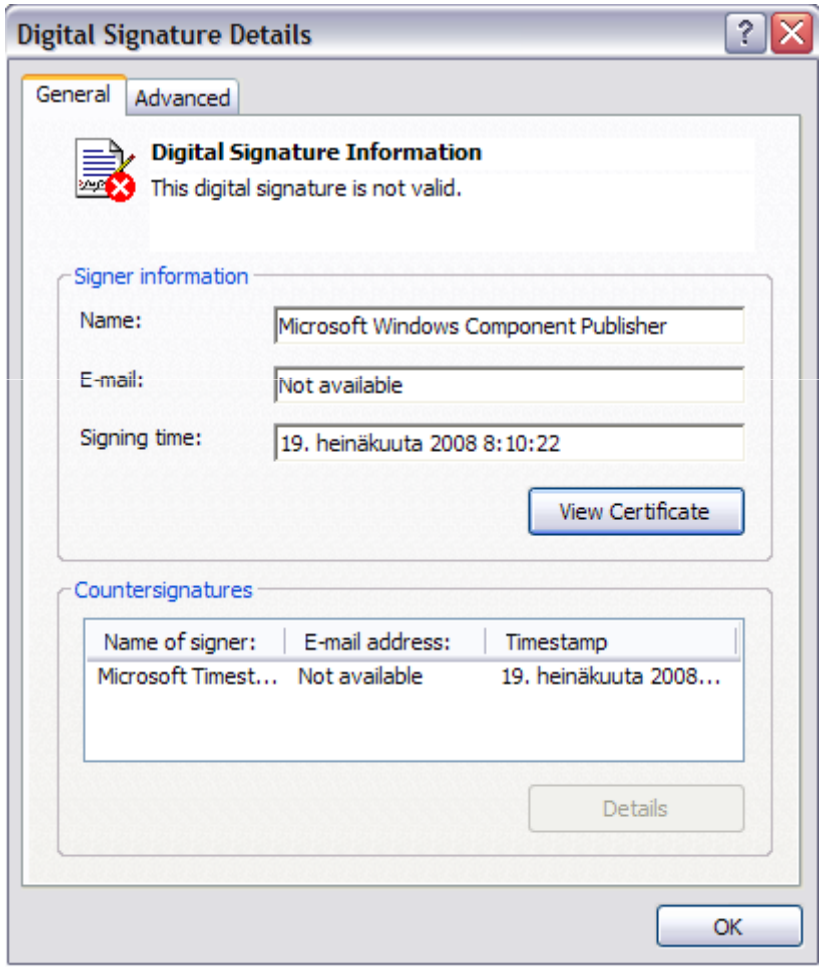
Copying Certificate information from clean files

- Simplest trick is to copy signature fields from clean files
  - Usually from Microsoft or well known security companies
  - Kaspersky and Symantec seem to be very popular for some reason
- Authenticode check fails on these
- But unfortunately that is difficult for user to detect in Windows
  - Basic properties UI is very deceptive
  - Vista and 7 UAC confirmation dialog does alarm on broken sig
    - Only after execution attempt, which may lead to human misclassifying a sample
- Our guess is that malware authors copy certificates in order to confuse users or AV analysts that file is signed by trusted party

Properties dialog for malware with copied cert

Backdoor:W32/Hupigon.OLY





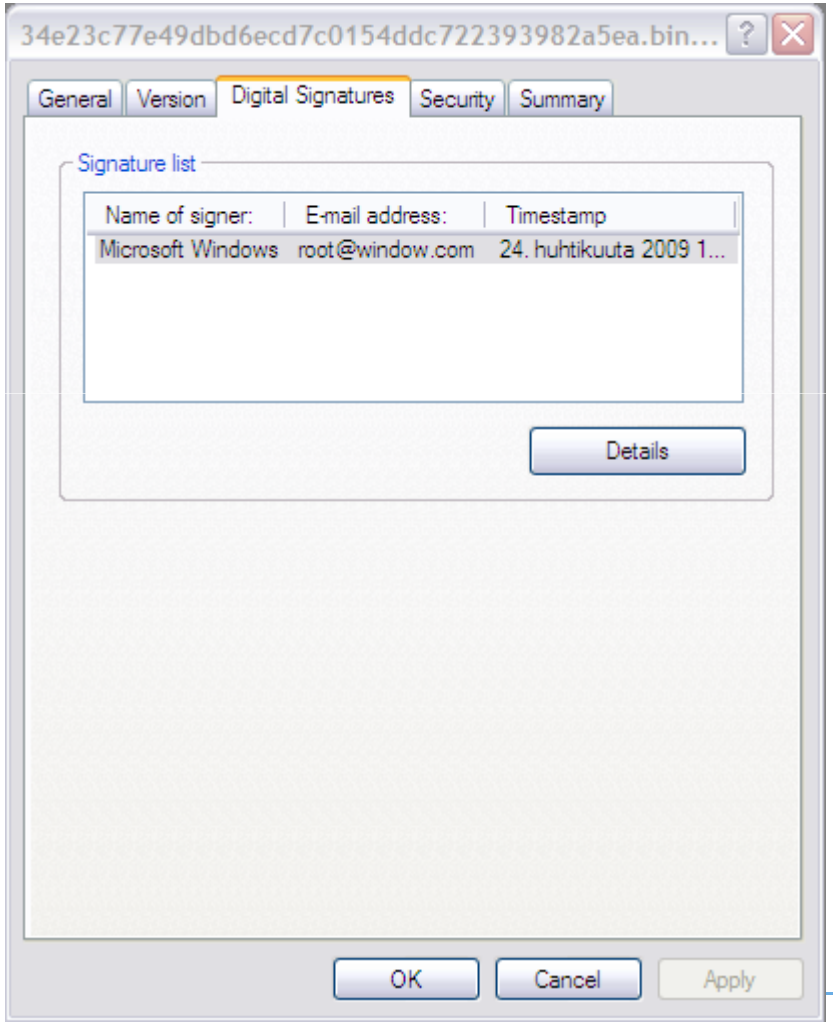
F-Secure

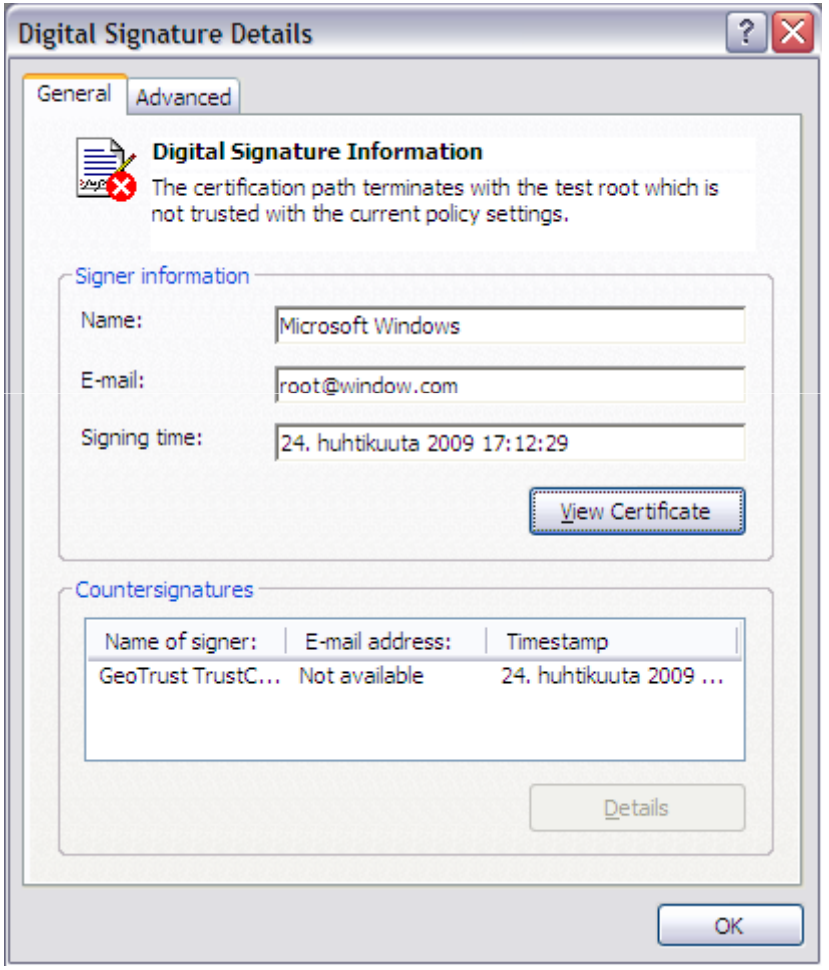
Selfsigned Certs With Fake Name

- Use fake Name Microsoft or other trustworthy company
- Windows signature check fails just like with copied cert
  - Properties dialog has same problem as with copied certs
- Tools that do not check CA validity will fail to detect these
  - Which can cause AV company to treat file as false alarm or require manual analysis on the file which causes much slower reaction
  - We have received FA reports on self signed files that are malware
  - Most likely whomever was checking the sample was fooled by self-signed cert

Typical Self Signed Cert Used By Malware

Trojan-Downloader:W32/Geral.AR





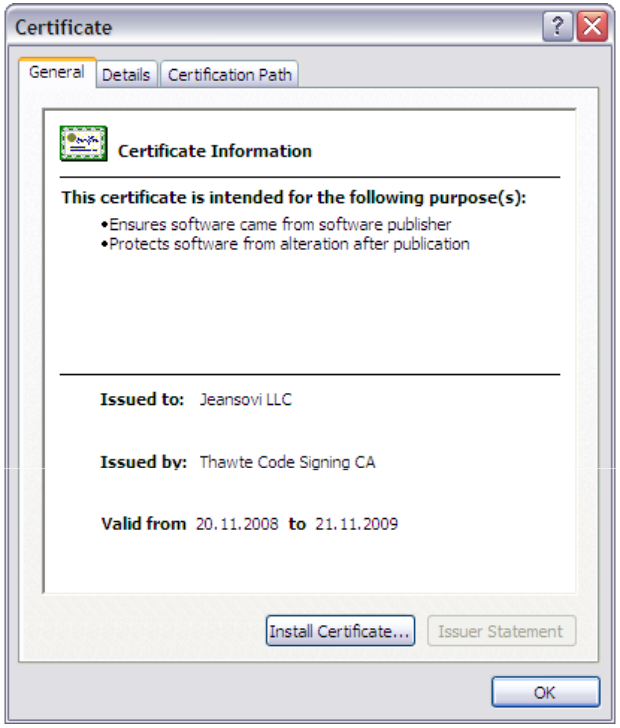
MD5 Forgery

- Unfortunately MD5 is still supported in code signing
- Weakness of MD5 in code signing is well demonstrated
  - In 2007 Marc Stevens, Arjen K. Lenstra, and Benne de Weger produced two EXEs with identical MD5 but different behaviour **[1]**
  - In 2009 Didier Stevens created tool to copy authenticode signature from one file to another that has identical MD5 **[2]**
- However real life examples we have seen are not practical
  - Either the files are very small
  - Or they differ only in predefined locations that affect program flow
- So far we have not found any real life case or even file that would have significant size and significant content

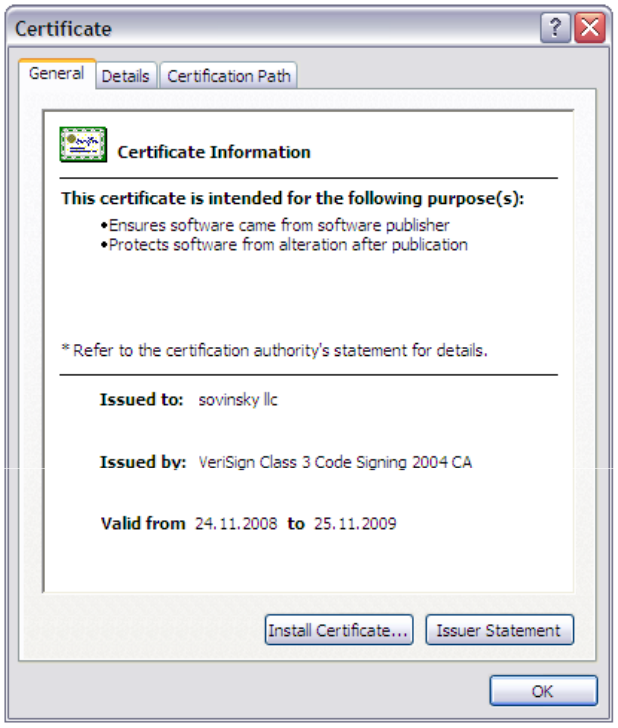
Get Certified And Be Evil

- As MD5 forgery is not feasible malware authors need certificates
- Thus they need to get valid cert from some CA
  - Most common way is just to get cert in valid company name
- Mostly used by riskware/potentially unwanted program authors
  - But also used lot by Rogue AV/Application companies
- Companies change name very frequently thus also their certs change
- For example “Perfect Defender “ is signed with following names
  - Jeansovi llc
  - Perfect Software llc
  - Sovinsky llc
  - Trambambon llc

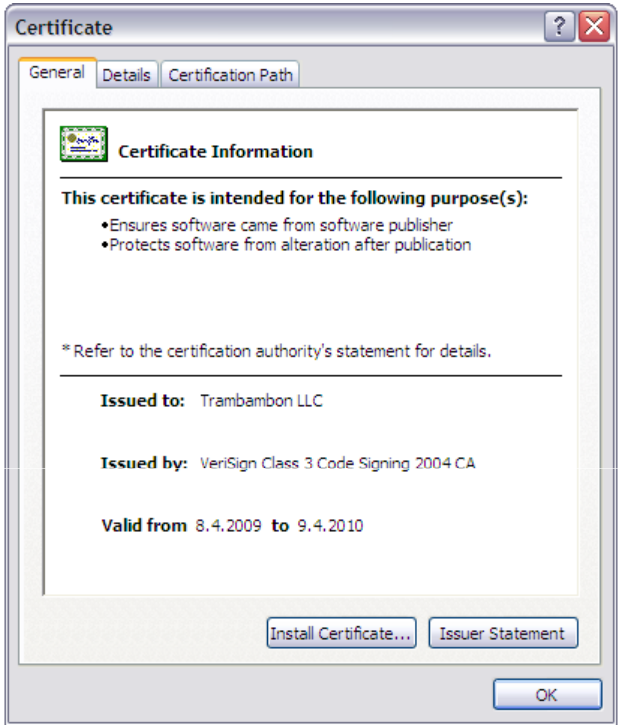
Perfect Defender Certificates



15bbb50ba1b5e532ed2c181b59e4c35714baf292

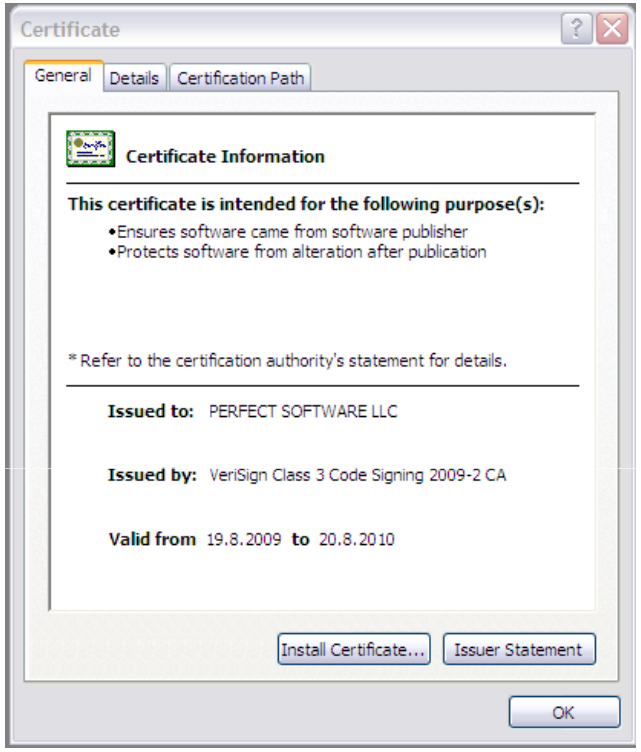


a43bece41cc5fbd631f52134de8b25f6159da60c



efc4894c06c2792ef78233387f98ad901e9d117a

Perfect Defender Certificates

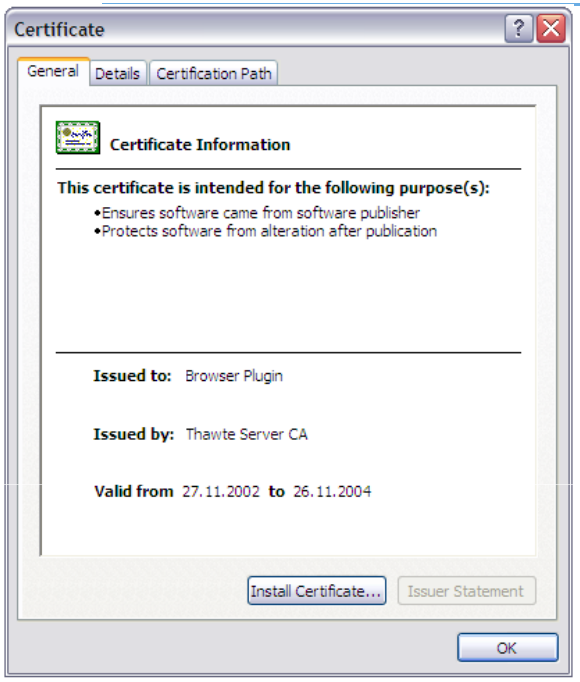


9a7875fe271930acf5018bbfaddebf6306f1dd78

Certificates With Misleading Name

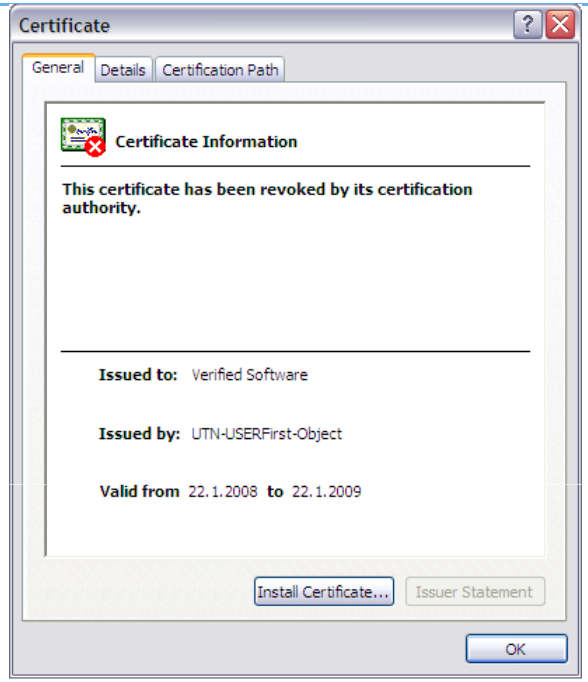
- CAs are supposed to give certificates only to valid companies
  - Malware can get valid name for a new company
  - But unknown company does not inspire trust in user
- What would user do if he sees dialogs with
  - Verified Software
  - Genuine Software Update Limited
  - Browser plugin
- Yes, these are real CA issued certificates
  - Examples I found are either expired or revoked
  - But certs like following examples should not have ever been issued

Would You Trust These?



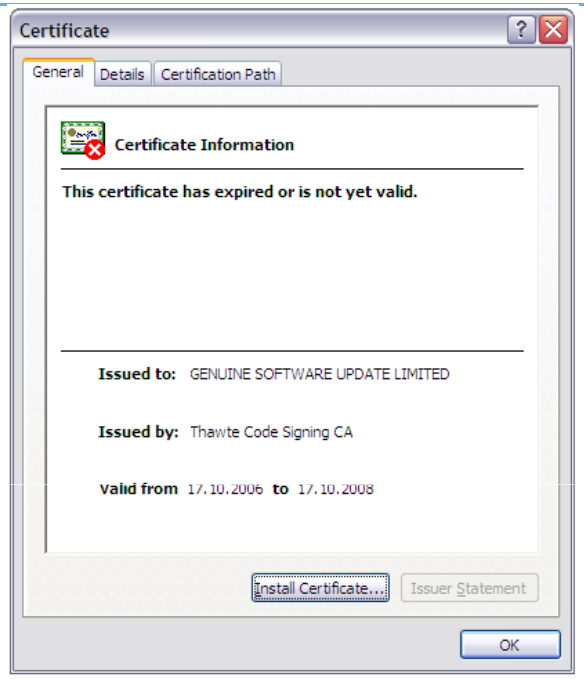
Trojan-Dropper:W32/Agent.DJDO

351e27c7edfdb121eff71eb2fd617f40318dd0a8



Rogue:W32/XPAntivirus.gen!E

865bc9932290619009467b0546f8813dd0cdbf15



Trojan-Dropper:W32/Agent.DJDP

6f13c37af27c42f65af938c942dcf7f0762300d9

Get Cert On Someone Else's Name

- "Verified Software" will be quickly revoked when CA is notified
- Malware authors may try to get certs with real names
  - Names that have verifiable online reputation
- Just like anyone else, CAs automate to cut costs
  - Which can make their process vulnerable to fraud
  - We have seen researchers getting certs with names like Microsoft **[3]**
  - So getting cert in less critical name seems rather likely
- However CAs claim that they have very strict verification policies

Just How Good Those Policies Are?

- In May 2010 Kurt Seifired made research on CA verifications **[4]**
  - Some CAs, such as RapidSSL, treat email address as verification
  - If you can receive mail to admin address and click link you own that domain. Right?
  - What if the domain belongs to webmail and have one of following?
    - admin, administrator, hostmaster, info, is, it, mis, postmaster, root, ssladmin, ssladministrator, sslwebmaster, sysadmin, or webmaster@somedomain.com
- Some CAs may have similar loopholes for Authenticode certs
  - We did a survey where we asked developers about CA procedures
  - Email and simple paper check seems to be very common
  - Fortunately Kernel certs are more strictly vetted
    - So getting 64-bit Vista/Win 7 drivers signed is not that easy

Find Someone To Sign Stuff For You

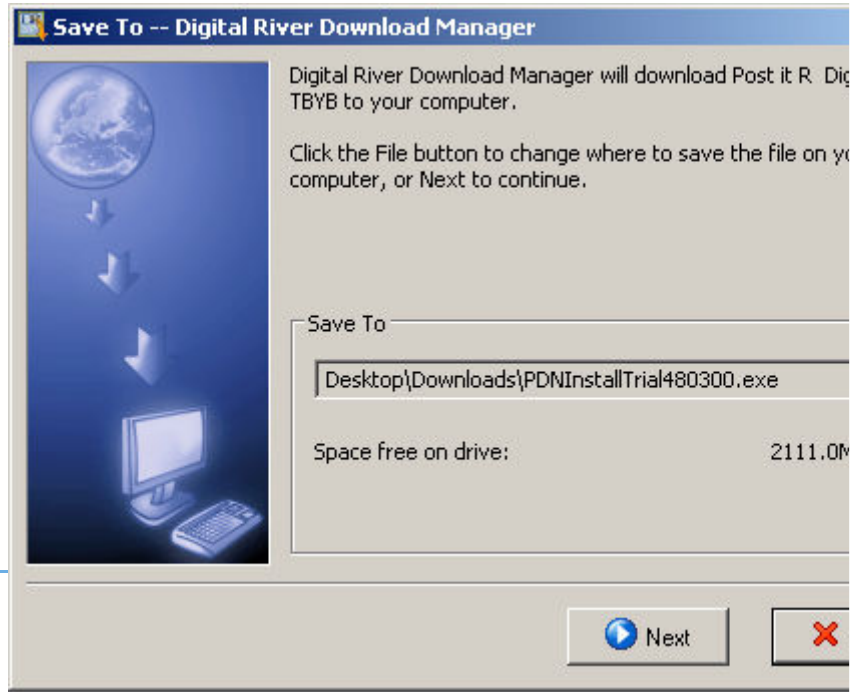
- Many in software industry view code signing as nuisance
- Thus their signing security can be lax and exploitable
- Some ecommerce operators sign binaries that they resell
  - As transaction processor is handling the software so putting their signature can make sense from their point of view
  - But unfortunately this gives a lot more credibility for arbitrary piece of software than it would otherwise have
- Code signing is supposed to be guarantee of authenticity
- Not just a stamp signifying that it is being sold through some transaction processor

Digital River

- One such transaction provider is Digital River (DR)
- DR is E-Commerce outsourcing company
  - In addition to typical services they sign binaries for their customers
- Currently our file collection has 55292 files signed by DR
  - Of which 295 are detected as rogues or malware
  - 3000+ as potentially unwanted
- DR signing services are currently used by rogues and PUPs
  - MSNSpyMonitor, WinFixer, QuickKeyLogger, ErrorSafe, ESurveiller
  - SpyBuddy, TotalSpy, Spynomore, Spypal

DR and GetRightToGo, What Ever Could Go Wrong?

- When researching Digital River we found an interesting set
- Downloaders built with GetRightToGo and signed by DR
  - They download and execute from third party URL
- As far as we can see DR, has no control what is downloaded from the URL, but they still give their “guarantee” for it
- Samples we checked downloaded clean screensavers, but these could be easily be used for evil



Steal Authenticode Private Key

- Stealing Authenticode keys would be obvious move
  - But we have not seen this approach in widespread use yet
- There are malware families that steal certs
  - Adrenalin bot kit
  - Ursnif family
  - Zeus family
- Malware authors have potential access to Authenticode keys
  - But we have not seen stolen certs being used yet
- Most likely this is due to Malware authors not having that big of a need for code signing just yet

Would There Be Useful Certs To Be Stolen

- We did a small survey to find out typical developer habits
- We got 69 answers
  - Which gives some indication but not definite conclusions
  - 69% Sign code on their development system
  - 45% Do not use password or have password in batch file
  - 87% Use their their development system for internet use
  - 12% Have had their development system infected in the past
- These results give ground for assumption that
  - If malware authors would need certs they could get them

Community Content

Our method to sign software with a certificate

The last paragraph of this page states, "Publishers use utility programs to sign the software they intend to publish." We use a batch file, the contents of which are

```
Rem This is Signit.Bat
Rem Usage: c:\vbprojects\ChgIt\signit FYChg_Consolidated
"C:\Program Files\Microsoft SDKs\Windows\v6.0A\Bin\SignTool.exe" sign /f "C:\VBProjects\Authenticode\BuenoSoftware.pfx" /p "P@ssw0rd"

pause
```

12/7/2009
Rhaiman

http://msdn.microsoft.com/en-us/library/ms537361.aspx

Infect Developers System

- Malware writers can try to infect developers system
  - And infect new files before they are signed
- Thus their malware would not only get signed by trusted certificate
  - But would also be distributed right in the application package
- We searched our collection for infections with valid signature
  - We found 548 Virus:W32/Induc.A infected samples
- So malware can get signed by developer
  - even when authors are not actively trying

What The Future Might Hold

- Current situation is still very easy for us
- So far malware authors have not had need to get signed
  - We have seen only rogues, individual cases and accidentally signed malware
- This will change with Windows 7
  - And unsigned software being treated with suspicion
- It is very likely that current trends will continue and get worse
  - Fooling CAs to give certs they should not issue
  - Developers being attacked for certificate theft
  - Developers being fooled to sign malware one way or another
  - Malware writers actively seeking rubber stamp channels like Digital River

What Should Be Done?

- Authenticode is too useful for us to ignore
  - We have to work as industry to prevent situation from getting worse
- Currently revocation processes are not working that well
  - Getting CAs to react on abuse reports requires a lot of work
    - Personally I have not received a single reply or reaction
- We need AV industry wide co-operation to fix this
  - We should have way to report compromised keys to each other
  - We should have common reporting channel to CAs
  - So that we do not have to fight through first level support when we report abuse case

Credits

- Ng Wah Keng
- Paul Dominic Anthony
- Mikko Suominen
- Kimmo Kasslin
- Mikko Hyppönen
- Mika Ståhlberg
- Toni Koivunen
- And everyone else at F-Secure labs who helped with clues and fact checking
- Nico Giansanti
- Marko Thure
- Mikko Hyykoski
- Sean Sullivan

References

1. http://www.mscs.dal.ca/~selinger/md5collision/
2. http://blog.didierstevens.com/2009/01/17/playing-with-authenticode-and-md5-collisions/
3. http://www.microsoft.com/technet/security/bulletin/MS01-017.mspx
4. http://www.linux-magazine.com/Issues/2010/114/BREACH-OF-TRUST

Protecting the irreplaceable

F-Secure