

Token

Kidnapping's

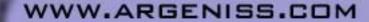
Revenge

Cesar Cerrudo

Argeniss

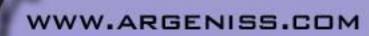
Who am I?

- Argeniss Founder and CEO
- I have been working on security for +8 years
- I have found and helped to fix hundreds of vulnerabilities in software such as MS Windows, MS SQL Server, Oracle Database Server, IBM DB2, and many more...
- +50 vulnerabilities found on MS products (+20 on Windows operating systems)
- I have researched and created novel attacks and exploitation techniques



Agenda

- Introduction
- What is impersonation and what are tokens?
- Windows XP and 2003 services security
- Windows 7, Vista and 2008 services security
- Token Kidnapping's revenge time
- Conclusions



Introduction

- In the past all Windows services ran as Local SYSTEM account
 - Compromise of a service==full system compromise
- Then MS introduced NETWORK SERVICE and LOCAL SERVICE accounts
 - Compromise of a service!=full system compromise
- Windows Vista, Windows 2008 and Windows 7 introduced new protections
- First Token Kidnapping issues were fixed, but as we are going to see Windows is still not perfect...

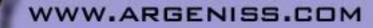
What is impersonation and what are tokens?

- Impersonation is the ability of a thread to execute using different security information than the process that owns the thread
 - ACL checks are done against the impersonated users
 - Impersonation APIs: ImpersonateNamedPipeClient(),ImpersonateLoggedOnUser(), RpcImpersonateClient()
 - Impersonation can only be done by processes with "Impersonate a client after authentication" (SeImpersonatePrivilege)
 - When a thread impersonates it has an associated impersonation token



What is impersonation and what are tokens?

- Access token is a Windows object that describes the security context of a process or thread
 - It includes the identity and privileges of the user account associated with the process or thread
 - -They can be Primary or Impersonation tokens
 - Primary are those that are assigned to processes
 - Impersonation are those that can be get when impersonation occurs
 - Four impersonation levels: SecurityAnonymous, SecurityIdentity, SecurityImpersonation, SecurityDelegation



Windows XP and 2003 services security

- Services run under Network Service, Local Service, Local System and user accounts
 - All services can impersonate
- Fixed weaknesses
 - A process running under X account could access processes running under the same X account
- After fixes
 - RPCSS and a few services that impersonate SYSTEM account are now properly protected
 - WMI processes are protected now

Windows Vista, 2008 and 7 services security

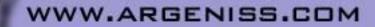
- Per service SID (new protection)
 - Nice feature, now service processes are really protected and its resources can be armoured
- Fixed weaknesses in Windows Vista and 2008
 - While regular threads were properly protected, threads from thread pools were not
 - WMI processes running under LOCAL SERVICE and NETWORK SERVICE were not protected
- After fixes
 - Threads from thread pools are properly protected
 - WMI processes are protected now

Token Kidnapping's revenge time

- First I found that Tapi service had process handles with duplicate handle permissions
- Then I started to examine the Tapi service
 - Found weak registry permissions
 - HKLM\SOFTWARE\Microsoft\Tracing
 - HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\T elephony
 - Found lineAddProvider() API, Network Service
 and Local Service accounts can load arbitrary dlls
 - Tapi service runs as System in Windows 2003
 - Found that Tracing functionality is used by most services, including services running as System

Token Kidnapping's revenge time

- Previous findings lead to other interesting findings in Windows 2003
 - When WMI is invoked, DCOMLaunch service reads Network and Local Service users registry keys
 - If values are found then HKCR keys are not used
 - Allows WMI process protection bypass
- Finally I could elevate privileges from Local/Network Service in all Windows versions and bypass protections



Token Kidnapping's revenge time

- Windows 2003 IIS 6 & SQL Server exploits
 - -Bypass WMI protection
- Windows 2008 and Windows 7 IIS 7.5 exploits
 - Exploit weak registry permissions



Recomendations

- On IIS don't run ASP .NET in full trust and don't run web sites under Network Service or Local Service accounts
- Avoid running services under Network Service or Local Service accounts
 - Use regular user accounts to run services
- Remove Users group from HKLM\Software\Microsoft\Tracing registry key permissions
- Disable Telephony service

Conclusions

- New Windows versions are more secure but there are still some issues easy to find
- Finding vulnerabilities is not difficult if you know what tools to use and were to look for
- On Windows XP and Windows 2003
 - If a user can execute code under Network Service or Local Service account
 - User can execute code as SYSTEM
- On Windows 7, Vista and 2008
 - If a user can impersonate
 - User can execute code as SYSTEM

References

- Token Kidnapping
- http://www.argeniss.com/research/TokenKidnapping.pdf
- Impersonate a client after authentication
- http://support.microsoft.com/kb/821546
- Access tokens
- http://msdn2.microsoft.com/en-us/library/aa374909.aspx
- Process Explorer and Process Monitor
- http://www.sysinternals.com
- API Impersonation Functions
- http://msdn.microsoft.com/enus/library/cc246062(PROT.10).aspx



Fin

- Questions?
- Thanks
- Contact: cesar>at<argeniss>dot<com

Argeniss

WE BREAK ANYTHING

www.argeniss.com