EADS

EADS INNOVATION WORKS

Pass The Hash

Nicolas RUFF

EADS-IW SE/CS

nicolas.ruff (à) eads.net

Plan

- Principe
- Historique
- Pass The Hash Toolkit 1.3
- Conclusion

Principe de l'attaque

- Dans la plupart des systèmes d'exploitation modernes, les mots de passe ne sont pas stockés "en clair" par le "service" d'authentification
  - Un hash (mathématiquement difficile à inverser) est stocké à la place
    - Windows LM : DES(constante, clé = mot de passe)
    - Windows NTLM : MD4(mot de passe)
  - Authentification (LM et NTLMv1)
    - Le serveur envoie un "défi"
    - Le client calcule F(défi, hash)
    - Le serveur fait le même calcul et compare le résultat obtenu

Principe de l'attaque

- Attaque "pass the hash" :
  - Dans ce schéma, la connaissance du hash est suffisante pour s'authentifier
- Précisions :
  - Windows n'est pas seul concerné
    - tous les protocoles basés sur le même schéma sont vulnérables
    - Kerberos 5 est vulnérable
  - Aucune interface graphique et/ou API exportée par Windows ne permet de s'authentifier par hash
  - NTLMv2 ajoute une authentification mutuelle
    - mais cela ne protège pas contre l'attaque

Historique

- 1997, Liste de discussion Samba
  - http://www.securityfocus.com/bid/233/info
- 1999, L0pht
  - Vulnérabilité de rejeu des défis pendant 15 minutes
  - http://www.sdnp.undp.org/rc/forums/tech/sdnptech/msg02200.html
  - Techniquement, ils possèdent toute la connaissance pour implémenter une attaque "pass the hash"
- 2001, SMBProxy (Cqure)
  - http://www.cqure.net/wp/?page_id=11
- 2001, SMBRelay (cDc)
  - http://en.wikipedia.org/wiki/SMBRelay
- 2003, SMBProxy (Foofus)
  - Patch pour Samba 3.x et Samba-TNG
  - http://www.foofus.net/jmk/passhash.html

Historique

- 2004, DreamPack PL
  - Patch en profondeur de Windows pour permettre l'authentification par hash depuis l'interface graphique
    - et autres joyeusetés
  - http://www.d--b.webpark.pl/dreampackpl_en.htm
  - Enfin une implémentation utilisable "sur le terrain"

- SSTIC 2007, Aurélien Bordes
  - http://actes.sstic.org/SSTIC07/Authentification_Windows/
  - Indique comment récupérer localement (et facilement) le hash d'un utilisateur connecté
    - Remarque : avant Windows XP SP2, une copie "en clair" du mot de passe était conservée en mémoire pendant toute la session utilisateur

Historique

- 2007, smbshell.nbin (Tenable Security)
  - Plugin Nessus permettant d'ouvrir un shell distant par hash
- 2007, Pass The Hash Toolkit (Hernan Ochoa / CORE Security)
  - http://oss.coresecurity.com/projects/pshtoolkit.htm
  - http://hexale.blogspot.com/
  - http://www.coresecurity.com/index.php5?module=ContentMod&action=item&id=1030

Pass The Hash Toolkit 1.3

- GenHash.exe
  - Calcule les hash LM et NTLM d'un mot de passe
    - Basé sur l'API Windows
      - SystemFunction006()
      - SystemFunction007()
    - Pratique pour tester le support des caractères "spéciaux"
    - Une implémentation "stand alone" est disponible depuis longtemps :
      - http://www.groar.org/groar/

Pass The Hash Toolkit 1.3

- IAm.exe
  - Permet de changer dynamiquement son identité Windows
  - Syntaxe :
    - iam.exe username domainname LMhash NThash
  - Permet d'utiliser ensuite tout type d'outil d'administration basé sur l'authentification Windows
    - Ex. SQL Management Studio, etc.

Pass The Hash Toolkit 1.3

- Détails :
  - Fonctionne par accès aux structures de LSASRV.DLL
    - LsapAddCredential()
    - LsaEncryptMemory()
    - g_Feedback
    - g_pDESXKey
    - struct _LSAP_LOGON_SESSION *LogonSessionList
    - unsigned long LogonSessionCount

Pass The Hash Toolkit 1.3

- Base de "signatures" pour trouver les structures susmentionnées
  - Un script ".idc" est fourni pour trouver les adresses nécessaires
  - Il existe également une heuristique basée sur les premiers octets de chaque fonction
- Requière les droits "administrateur"
  - Pour ouvrir le processus "LSASS.EXE"
- Active le privilège SeDebug
  - Pour injecter "iamdll.dll"

Pass The Hash Toolkit 1.3

- Les hash sont chiffrés en mémoire
  - LsaEncryptMemory() = DESX()
  - Clé de chiffrement initialisée dans :
    - » LsaInitializeProtectedMemory()
  - Par :
    - » ulong g_cbRandomKey = 256
    - » uchar * g_pRandomKey
    - » SystemFunction036() = RtlGenRandom()
    - » Fonction récemment attaquée par des chercheurs ☺

Pass The Hash Toolkit 1.3

- WhosThere.exe
  - Permet d'énumérer les sessions ouvertes sur la machine et les hash associés
  - Détails :
    - Même base de signatures que "iam.exe"
    - Requière les droits "administrateur"
      - Pour ouvrir le processus "LSASS.EXE"
    - Active le privilège SeDebug
      - ReadProcessMemory()
    - Les hash sont stockés dans chaque "LogonSession"

Pass The Hash Toolkit 1.3

- Forces
  - Des outils qui marchent "pour de vrai"
  - Vista supporté
  - Code source disponible
  - Pas ou peu détecté par les antivirus

- Faiblesses
  - Windows 2000 pas (encore) supporté
  - Kerberos pas (encore) supporté

Conclusion

- L'attaque "pass the hash" est connue depuis longtemps
- Mais l'état de l'art a considérablement évolué
  - Partage considérable d'informations (Samba, Wine, ReactOS)
  - Outils d'analyse statique et dynamique plus puissants
- Désormais, il existe des outils qui marchent
- Impacts de ces attaques :
  - Complexité des mots de passe sans effet si la base est compromise
  - Rebond sur un poste compromis plus facile
- Protection : aucune ?