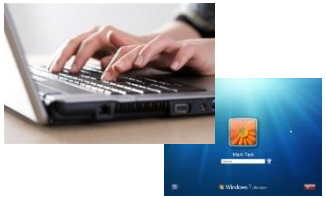
Post-Explotación con WCE v1.2

Pass-the-Hash. Pass-the-ticket y más…

Fecha: 0**1-07-2011**

Autor:
Hernan Ochoa
<hernan@ampliasecurity.com>

Autenticación en Windows



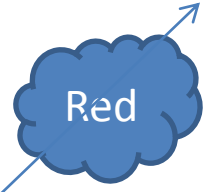
cgarcia
clave1

h1 = LMHash("clave1")
h2 = NTHash("clave1")

cmp([h1,h2], SAM)

SAM Database

Usuario	Hash LM	Hash NT
cgarcia	A6BCD..	B0FD1..
ajuarez	B90DF..	CCDF..
…	…	…



- **No se envía la clave plana**
- **Se usa el protocolo NTLM**

Autenticación NTLM

lmhash = LMHash("clave1")
nthash = NTHash("clave1")

Inicia conexión

Responde *C* = challenge random

Envía *cgarcia*, *R*

cgarcia
clave1

R = f(*lmhash*/*nthash*, *C*)

SAM Database

Usuario	Hash LM	Hash NT
cgarcia	A6BCD..	B0FD1..
ajuarez	B90DF..	CCDF..
…	…	…

R' = f(SAM[lmhash/nthash], C)
R' == R => Acceso Autorizado
R' <> R => Acceso Denegado

Ataques pre-*Pass-the-Hash*

- Luego de comprometer una máquina Windows…
  - ‘ Dumpear’ la SAM
    - *pwdump3/3e/4/5/6/7,fgdump*,etc
      - Administrator:500:010203040506070809 0A0B0C0D0E0F10:1020304050607080 90A0B0C0D0E0F10
  - Crack/Brute-Force de los hashes para obtener la clave en cleartext
    - Requería tiempo…
    - No había garantía de obtener la clave
    - Rainbow tables no existían como hoy en día
      - Existía menor poder de cómputo, almacenamiento, etc
      - La técnica no era popular como hoy en día

Técnica Pass-The-Hash

Publicada por Paul Ashton en 1997

http://www.securityfocus.com/bid/233/discuss

amplia security

Técnica *Pass-the-hash*

lmhash = LMHash("clave1")
nthash = NTHash("clave1")

Inicia conexión

Responde *C* = challenge random

Envía *cgarcia*, *R*

cgarcia
clave1

R = f(***lmhash/nthash***, *C*)

SAM Database

Usuario	Hash LM	Hash NT
cgarcia	A6BCD..	B0FD1..
ajuarez	B90DF..	CCDF..
…	…	…

R' = f(SAM[**lmhash/nthash**], C)
R' == R => Acceso Autorizado
R' <> R => Acceso Denegado

- La clave en plano no es necesaria para hacer NTLM
- Sólo el lmhash/nthash son necesarios
- No es necesario crackear/brute-forcear
- Se usan los hashes directamente

Técnica *Pass-the-hash: 'explotación'*

- *Smbclient* modificado (SAMBA)
  - *smbclient //192.168.1.120/c$ -U Administrator -p 4ECC0E7568976B7EAAD3B435B51404EE:551E3B3215FFD87F5E037B3E3523D5F6*
- *Samba-TNG*
- Muchos stacks *SMB+NTLM* implementados por terceros
  - *Python*, *Ruby*, *Java*, etc

Stacks *SMB+NTLM* de 3ros: Limitaciones

- Tienen funcionalidad limitada y parcial
- Siempre corren atrás de Windows
  - Nueva funcionalidad debe ser implementada mediante reverse engineering
    - Complejo, require un gran esfuerzo y tiempo
- No se puede usar las herramientas nativas de Windows
  - Piden usuario y clave, no hashes..

Entra, *Windows Credentials Editor…*

Qué es WCE?

- Herramienta para manipular sesiones de logueo de Windows
  - Crear, listar, borrar, cambiar
  - Obtener credenciales asociadas a sesiones
  - **Pass-the-hash (NTLM)**
  - Pass-the-ticket (Kerberos)

Ataques pre-WCE/Pass-the-hash

- Sin WCE..
  - Crack/Brute-force de los hashes para obtener la clave en plano para usar en Windows
  - Crack/Brute-force de los hashes encriptados (C,R->NTLM) para obtener la clave en plano para usar en Windows
  - Uso de stacks SMB+NTLM de 3ros, con funcionalidad limitada y parcial
  - Mas difícil/no posible hacer Pass-the-hash en escenarios de pivoteo entre máquinas Windows

Ataques post-WCE/Pass-the-hash

- Con WCE…
  - Hacer *Pass-the-hash* directamente con los hashes
    - No hace falta crack/brute-force de los hashes
  - Se eliminan problemas de los stacks SMB+NTLM de 3ros.
  - Mas fácil hacer *Pass-the-hash* en escenarios de pivoteo entre máquinas Windows

Demo #1
WCE: Pass-the-hash

WCE: ‘Robar’ Credenciales en memoria

- Nuevo ataque implementado por WCE
  - **No es** *Pass-the-hash*, es otra técnica..
  - Mucha gente lo confunde, pero no es lo mismo..

- Permite obtener nombres de usuario y hashes guardados en memoria..

WCE: ‘Robar’ Credenciales en memoria

- Por qué quedan en memoria?
  - NTLM auth package
  - SSO

cgarcia
clave1

MiDominio/cgarcia
h1 = LMHash(“clave1”)
h2 = NTHash(“clave1”)

Red

NTLM

WCE: ‘Robar’ Credenciales en memoria

Cuándo quedan en memoria?

- Sesiones interactivas en la consola
- Sesiones remotas RDP
- RunAs
- Servicios de Windows que usan cuentas de usuario
- APIs de Window usadas por programas
- Etc.

WCE: ‘Robar’ Credenciales en memoria

Cuándo quedan en memoria?

- Sesiones interactivas en la consola

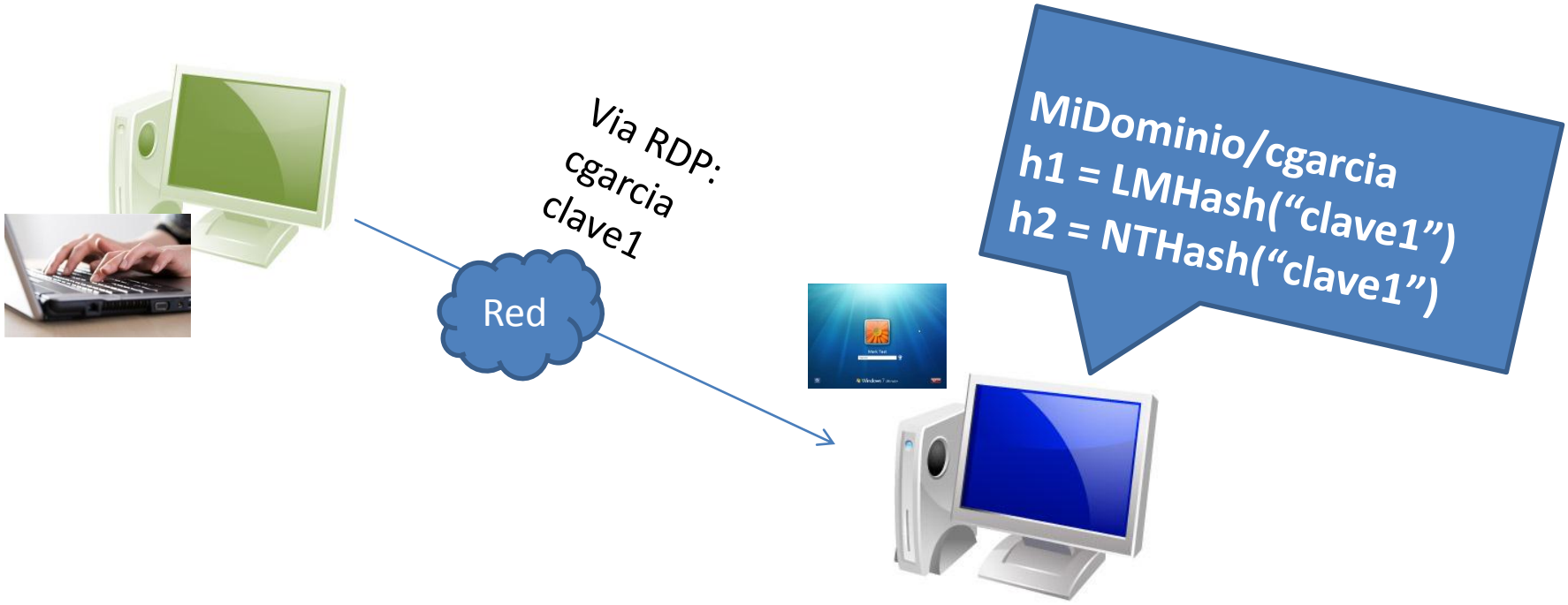
cgarcia
clave1

MiDominio/cgarcia
h1 = LMHash(“clave1”)
h2 = NTHash(“clave1”)

WCE: ‘Robar’ Credenciales en memoria

Cuándo quedan en memoria?

- Sesiones remotas RDP

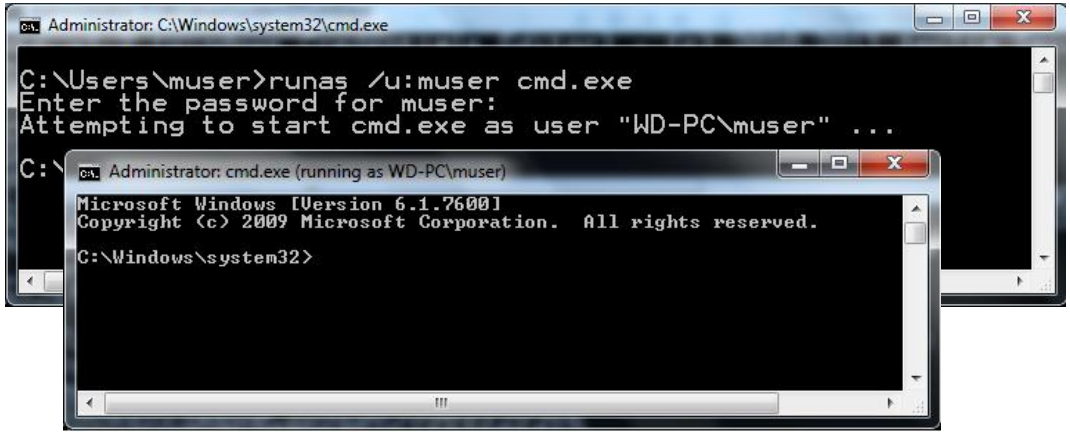


WCE: ‘Robar’ Credenciales en memoria

Cuándo quedan en memoria?

- RunAs

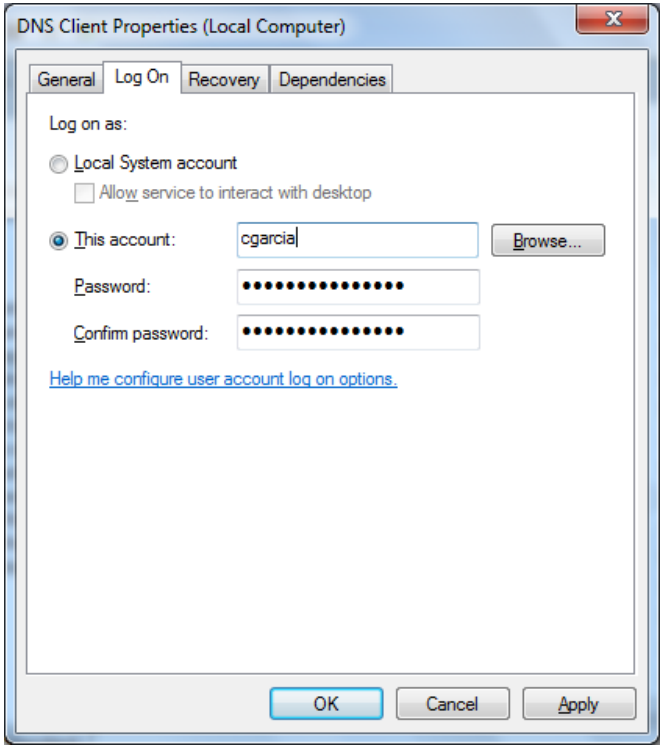
WD-PC/muser
h1 = LMHash(“clave4”)
h2 = NTHash(“clave4”)



WCE: ‘Robar’ Credenciales en memoria

Cuándo quedan en memoria?

- Servicios de Windows



WCE: ‘Robar’ Credenciales en memoria

Cuándo quedan en memoria?

- Programas que usan WIN32 APIs

Ejemplo:

LogonUser Function

msdn

The **LogonUser** function attempts to log a user on to the local computer. The local computer is the computer from which **LogonUser** was called. You cannot use **LogonUser** to log on to a remote computer. You specify the user with a user name and domain and *authenticate* the user with a *plaintext* password. If the function succeeds, you receive a handle to a token that represents the logged-on user. You can then use this token handle to impersonate the specified user or, in most cases, to create a *process* that runs in the context of the specified user.

Syntax

```
BOOL LogonUser(
  __in      LPTSTR lpszUsername,
  __in_opt  LPTSTR lpszDomain,
  __in      LPTSTR lpszPassword,
  __in      DWORD dwLogonType,
  __in      DWORD dwLogonProvider,
  __out     PHANDLE phToken
);
```

MiDominio/cgarcia
h1 = LMHash(“clave1”)
h2 = NTHash(“clave1”)

WCE: ‘Robar’ Credenciales en memoria

- WCE permite obtener el LM Hash..
  - Por defecto, hoy en día, Windows no guarda en la SAM el LM Hash
    - Porque es débil; “fácilmente” crackeable

WCE: ‘Robar’ Credenciales en memoria

- WCE puede obtener de memoria el LM Hash
  - Windows calcula y guarda los hashes en memoria, inclusive el LM hash
    - Sesiones interactivas
  - Posible crackearlo y obtener el plaintext para probar en lugares que no usan NTLM

WCE: ‘Robar’ Credenciales en memoria

Ejemplo:

El LM Hash no está en la SAM

```
Administrator:500:NO PASSWORD**********************:NO PASSWORD**********************
***:::
Guest:501:NO PASSWORD**********************:NO PASSWORD**********************:::
muser:1000:NO PASSWORD**********************:9BF617CAEFC9DFE18995B5A300174176:::
Completed.

C:\Users\muser\wce>
```

Salida de pwdump

El LM Hash está en memoria

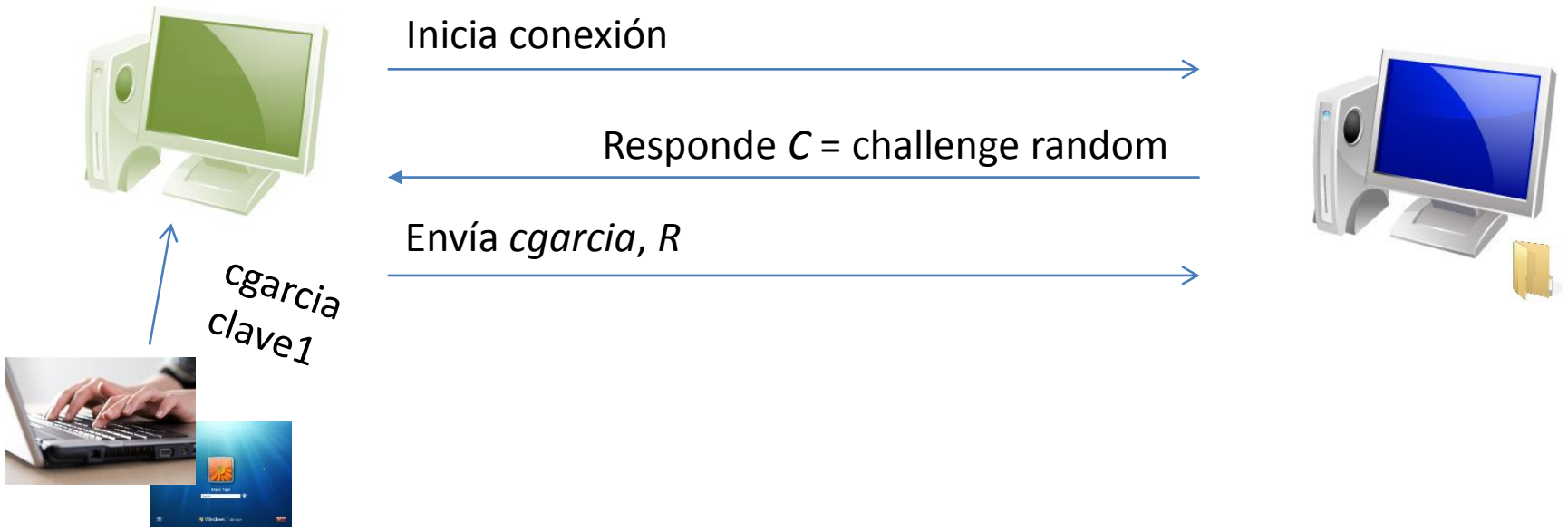
```
C:\Users\muser\wce>wce
WCE v1.2 (Windows Credentials Editor) - (c) 2010,2011 Amplia Security - by Herna
n Ochoa (hernan@ampliasecurity.com)
Use -h for help.

muser:wd-PC:A3283469F98CF766AAD3B435B51404EE:9BF617CAEFC9DFE18995B5A300174176

C:\Users\muser\wce>
```

Salida de WCE

WCE: 'Robar' Credenciales en memoria

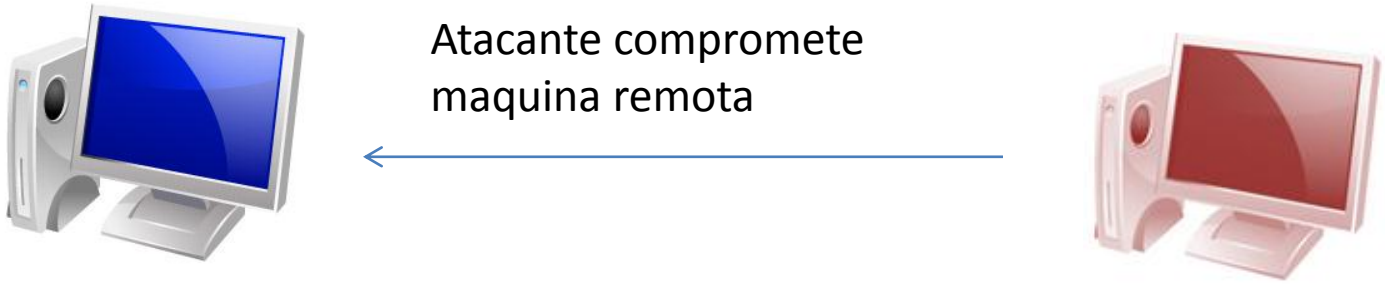
Cuándo **NO** quedan en memoria?

- Network Logons
  - El hash nunca llega al servidor remoto



WCE: ‘Robar’ Credenciales en memoria

Escenario de Ataque de post-explotación

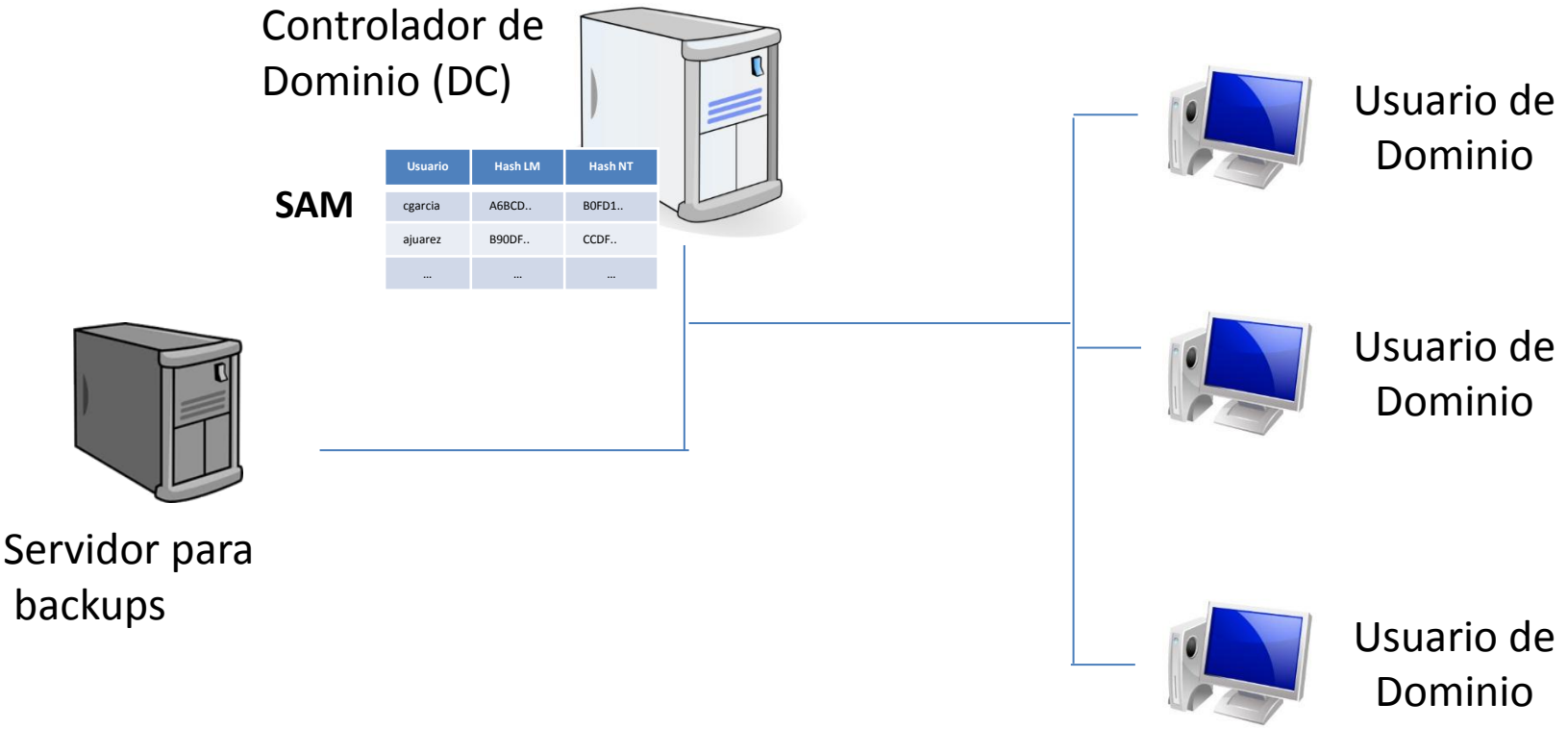


- Correr WCE para obtener credenciales en memoria
- Permite obtener hashes que pueden ser usados con WCE para hacer Pass-The-Hash

Demo #2
WCE: ‘Robar’ credenciales de memoria

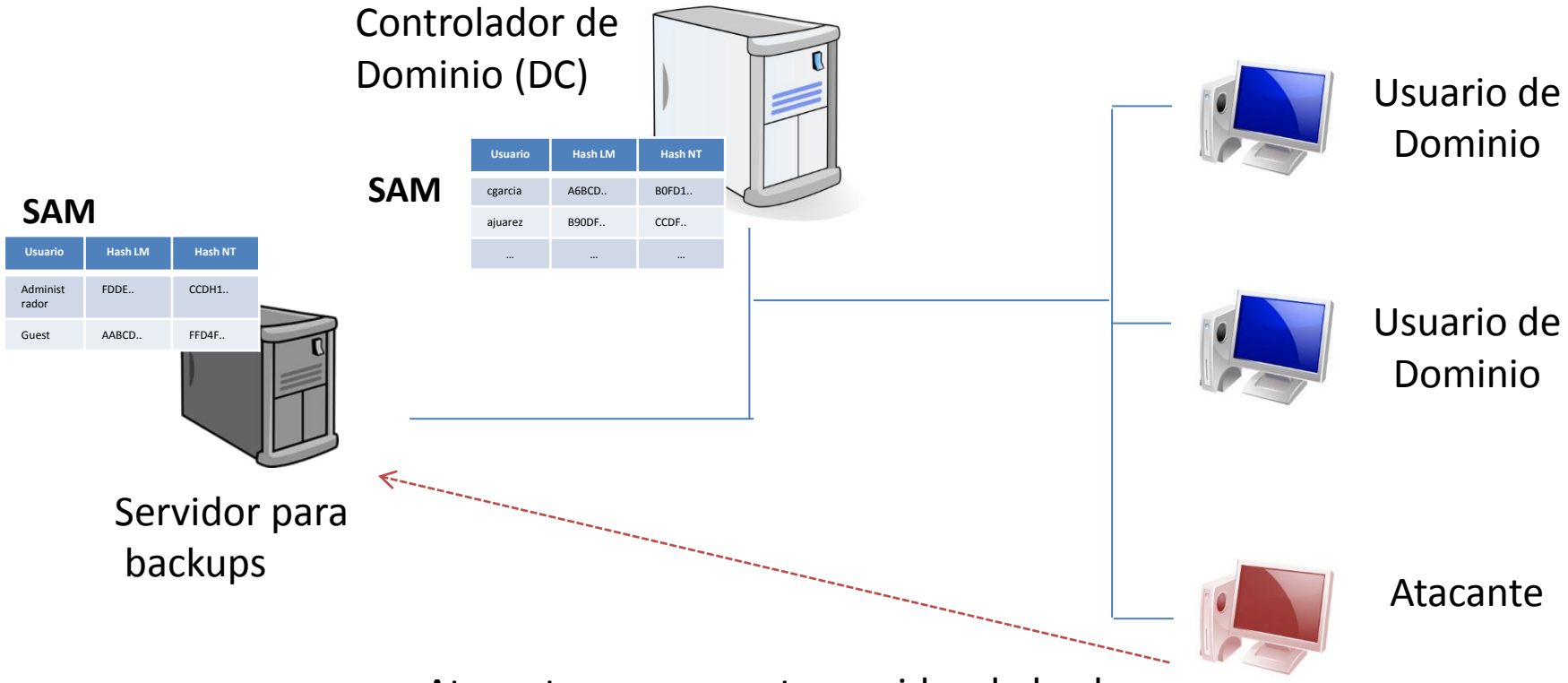
WCE: ‘Robar’ Credenciales en memoria

Especialmente interesante en Redes de Dominio Windows



WCE: ‘Robar’ Credenciales en memoria

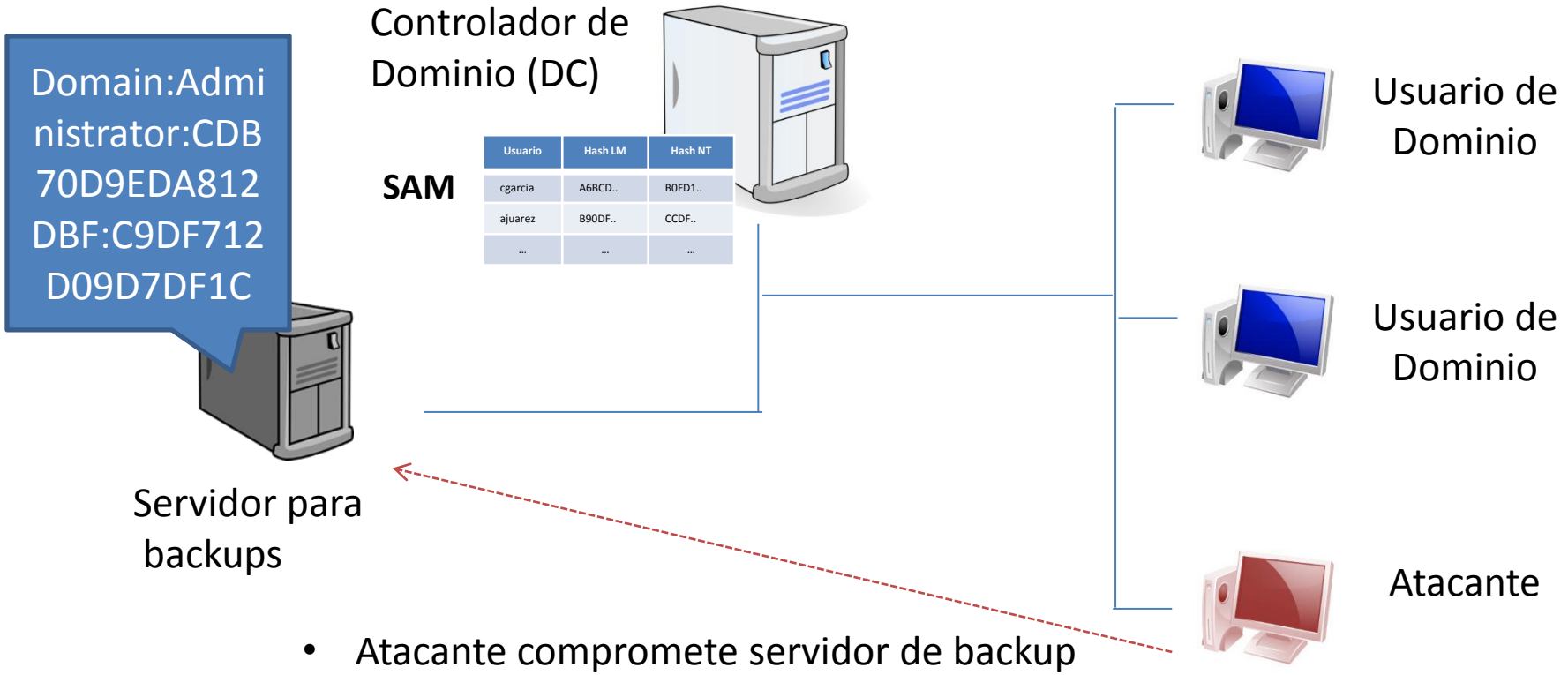
SIN WCE..



- Atacante compromete servidor de backup
- Solo tiene accesso a la SAM local..
- No hay usuarios de dominio, no muy útil

WCE: ‘Robar’ Credenciales en memoria

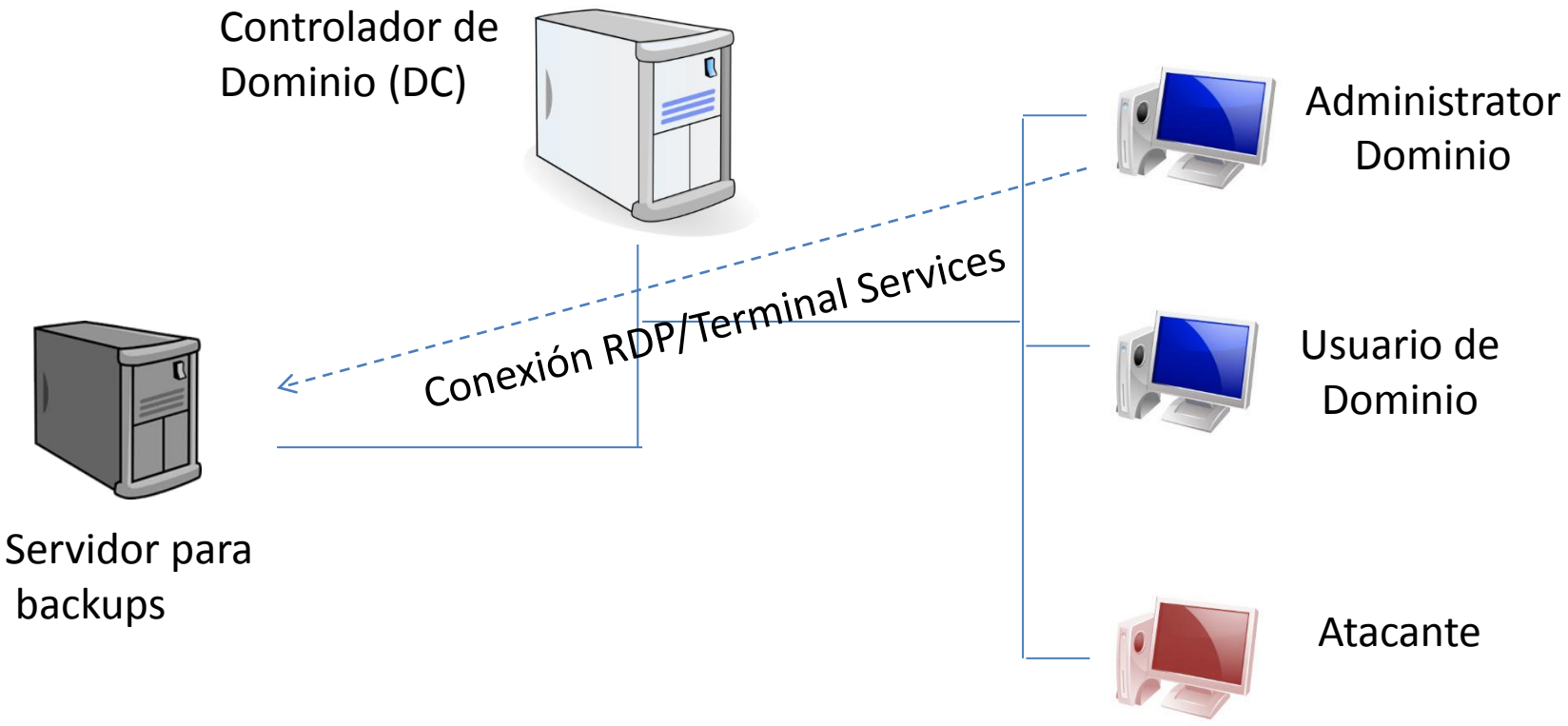
CON WCE..



- Atacante compromete servidor de backup
- Puede obtener credenciales de memoria..
  - Probablemente habrá usuarios de dominio
  - Posible compromiso de todo el dominio!

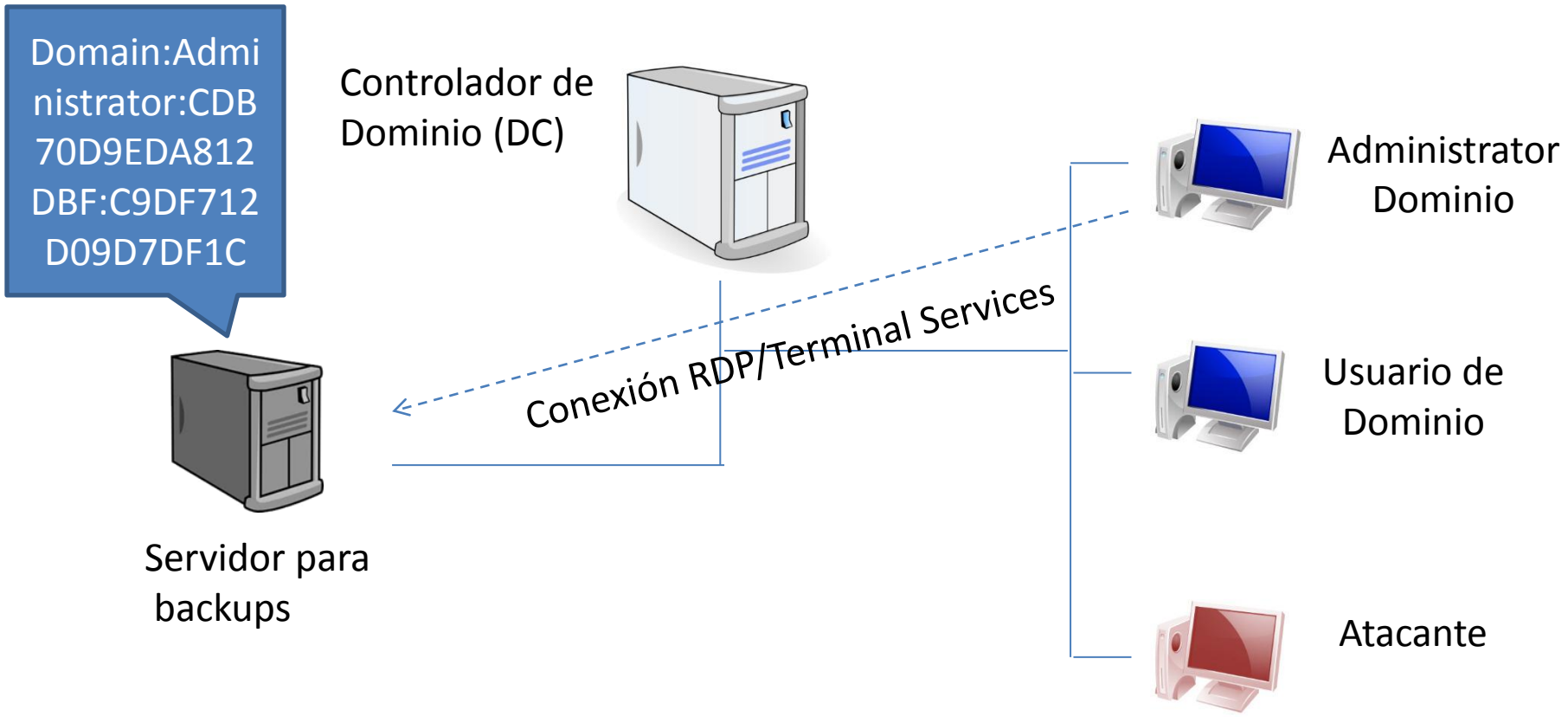
WCE: ‘Robar’ Credenciales en memoria

Escenario Tipo



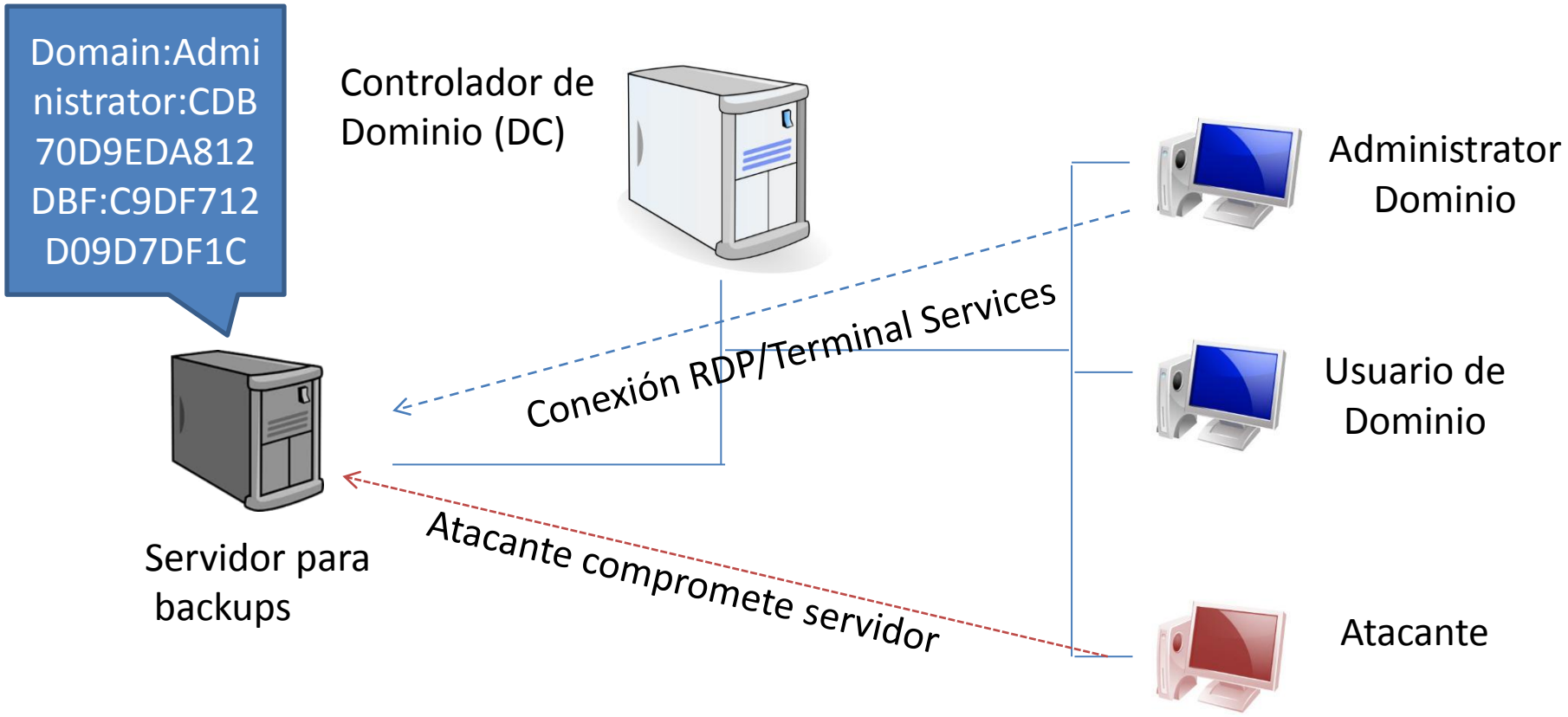
WCE: ‘Robar’ Credenciales en memoria

Escenario Tipo



WCE: ‘Robar’ Credenciales en memoria

Escenario Tipo



- **WCE ‘roba’ las credenciales del Administrator de Dominio!**

WCE: ‘Robar’ Credenciales en memoria

RDP ´expone´ credenciales

- **Cuando se hace RDP a una máquina remota, se dejan en su memoria los hashes NTLM de la clave**
  - Los NTLM hashes son equivalentes a la clave (pass-the-hash+wce)
  - Entonces, podría decirse que se está dejando la clave…

WCE: ‘Robar’ Credenciales en memoria

RDP ´expone´ credenciales

- **Aún cuando se usa pass-through authentication**
  - Las credenciales (user+domain+hashes) están guardadas en el Domain Controller
  - Pero al hacer RDP quedan también en la memoria de la máquina remota!
  - Cuidado a qué computadora se conecta usando RDP…

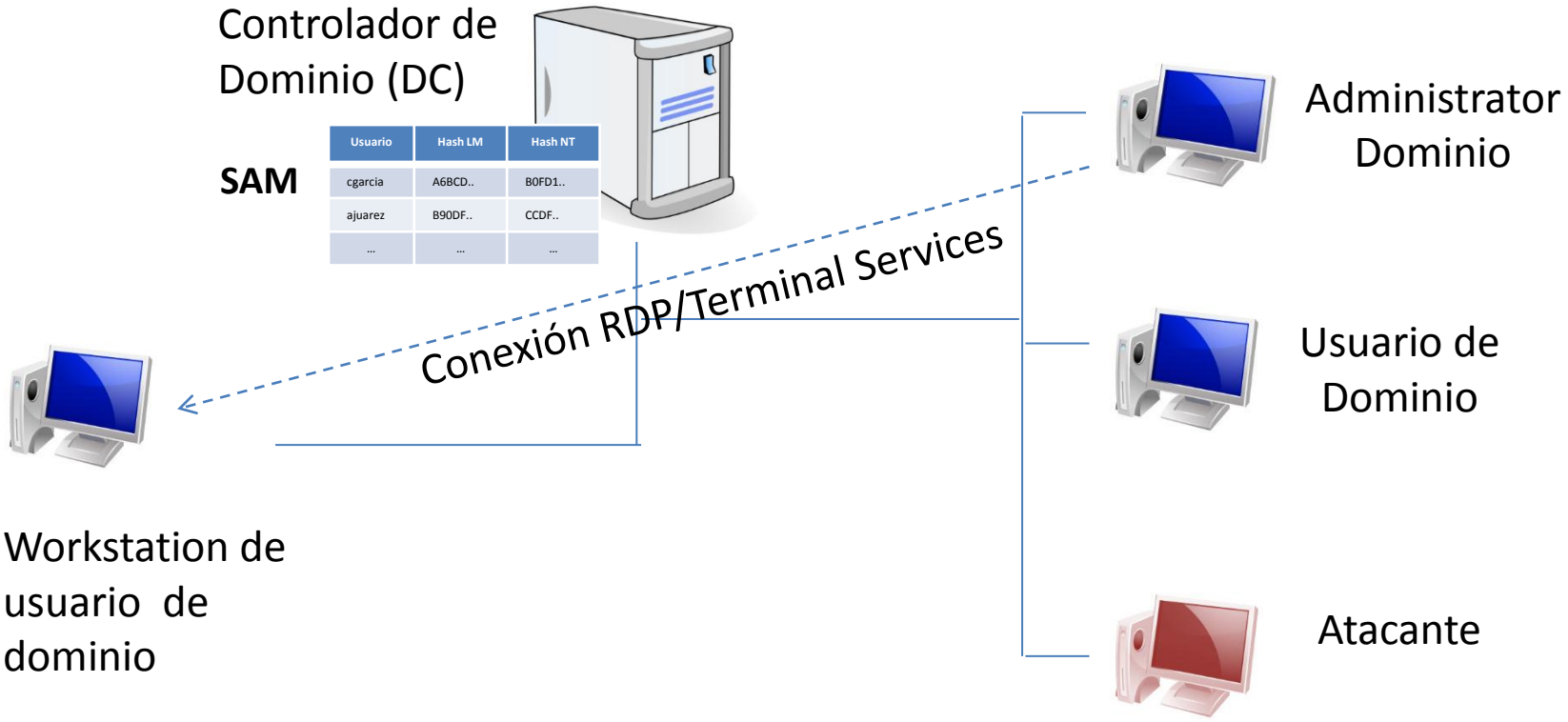
WCE: ‘Robar’ Credenciales en memoria

RDP ´expone´ credenciales

- Un atacante en la máquina remota puede obtener las credenciales leyendo la memoria
  - Administrador Local
  - Usuario de Dominio normal con privilegios de administrador local
  - Atacante que compromete workstation menos segura, usuario normal mas vulnerable
  - Etc.

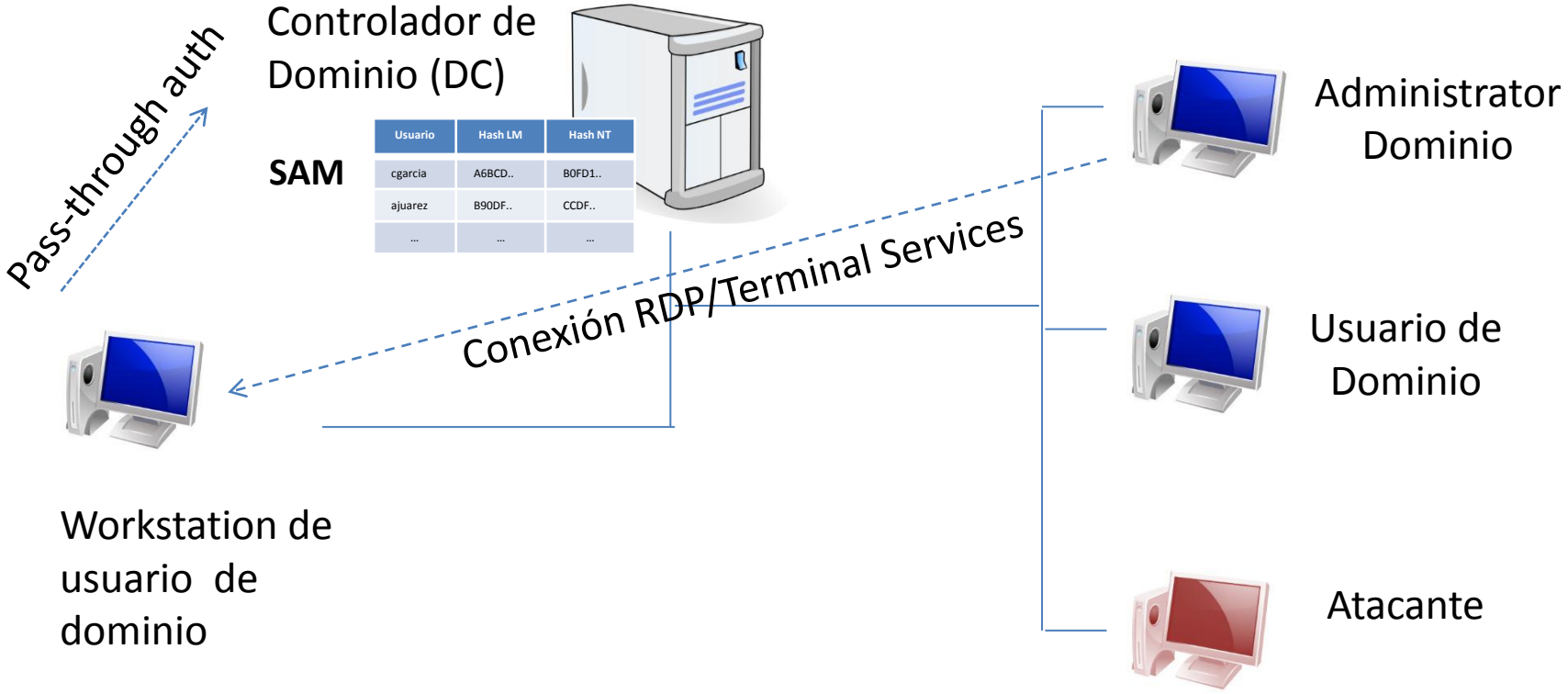
WCE: ‘Robar’ Credenciales en memoria

RDP ´expone´ credenciales: Ejemplo



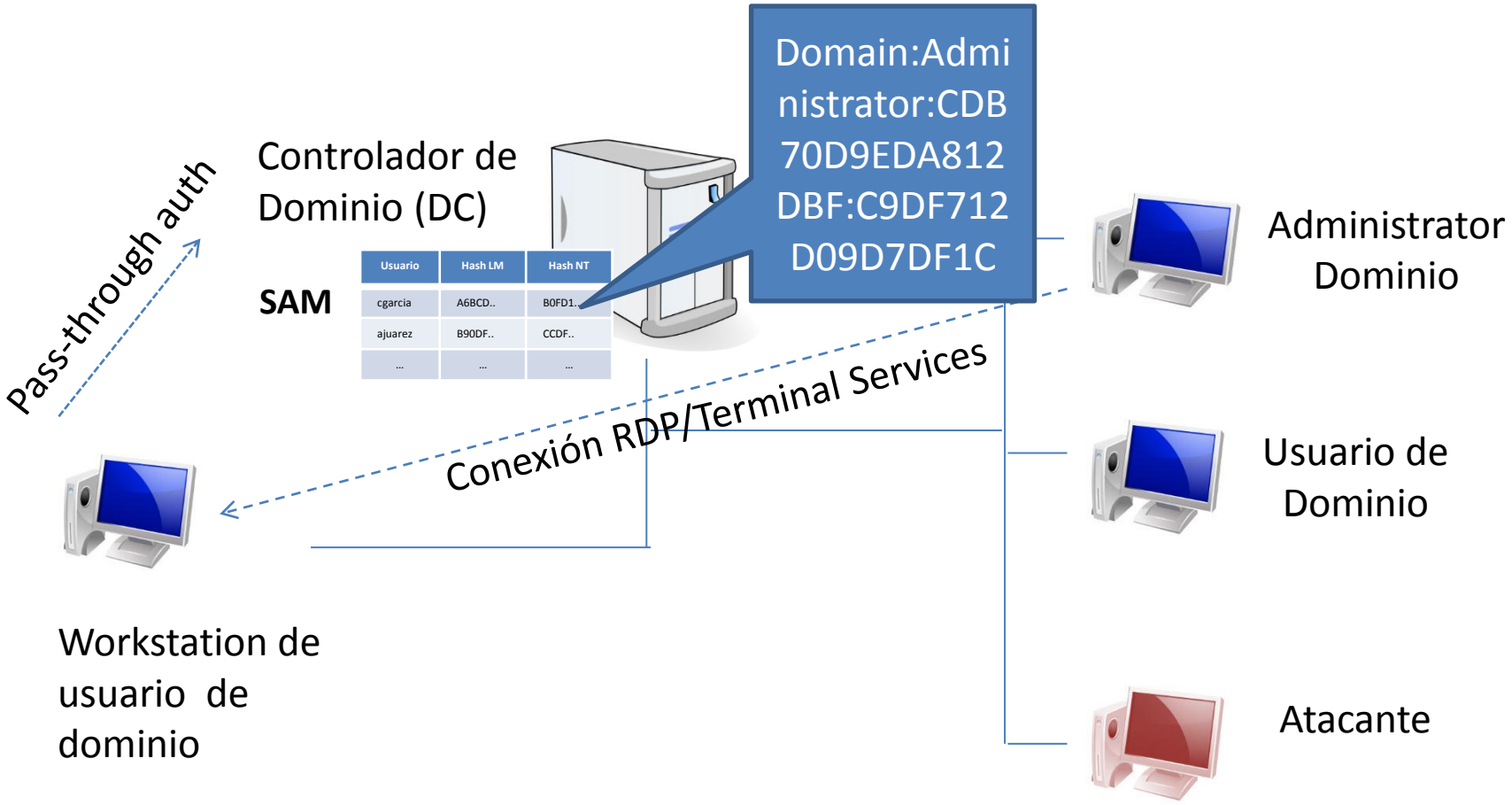
WCE: ‘Robar’ Credenciales en memoria

RDP ´expone´ credenciales: Ejemplo



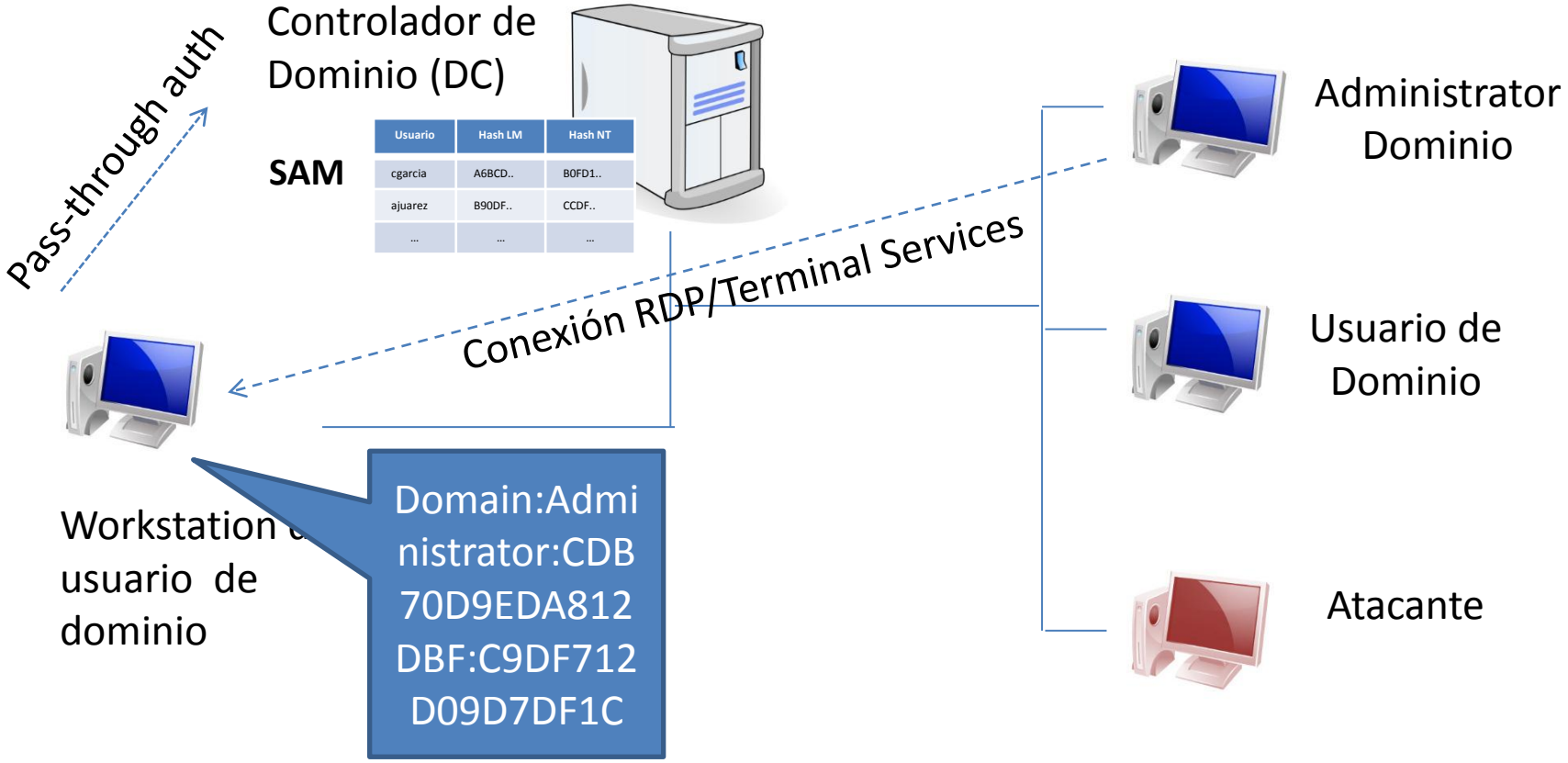
WCE: ‘Robar’ Credenciales en memoria

RDP ´expone´ credenciales: Ejemplo



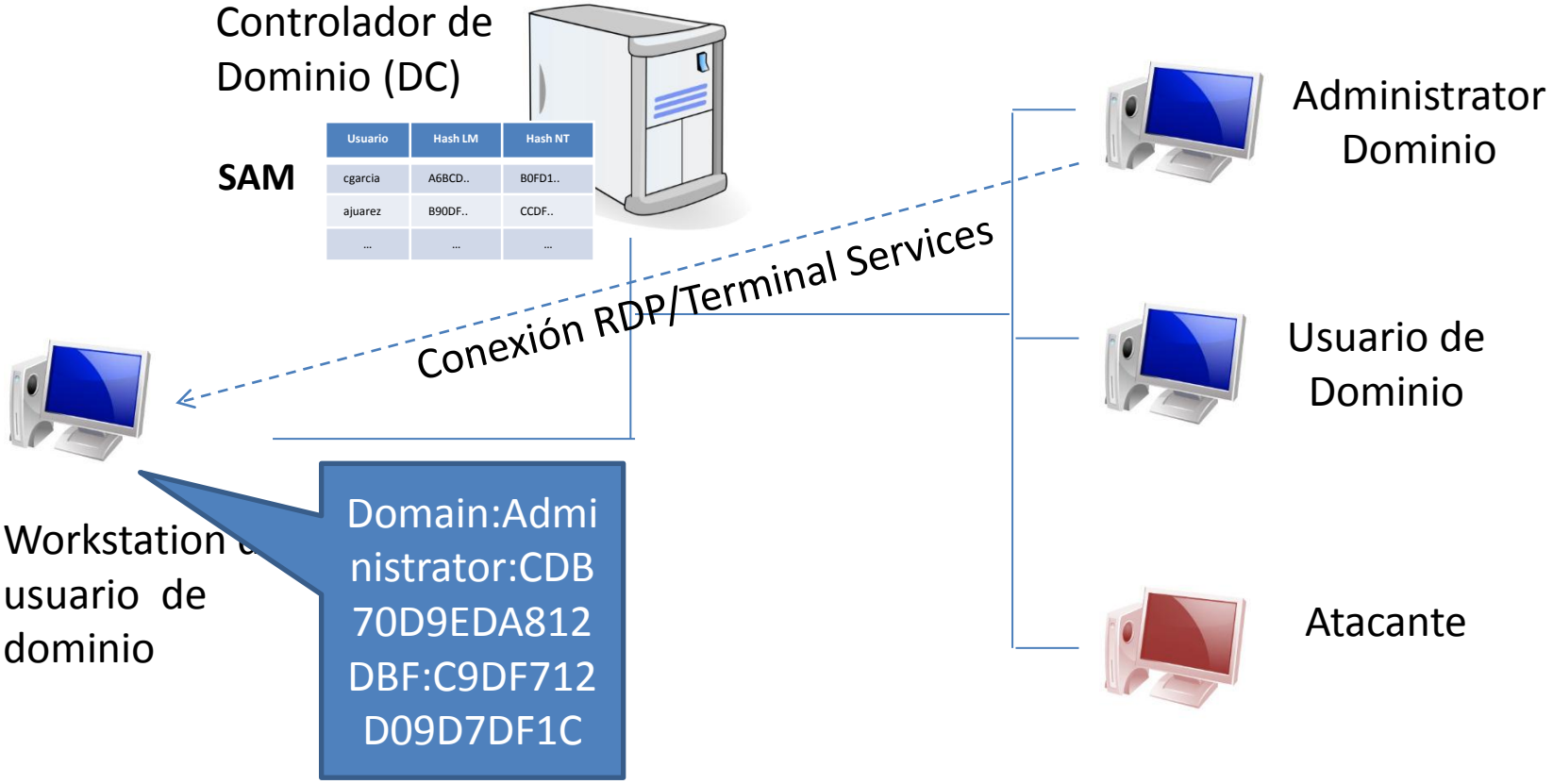
WCE: ‘Robar’ Credenciales en memoria

RDP ´expone´ credenciales: Ejemplo



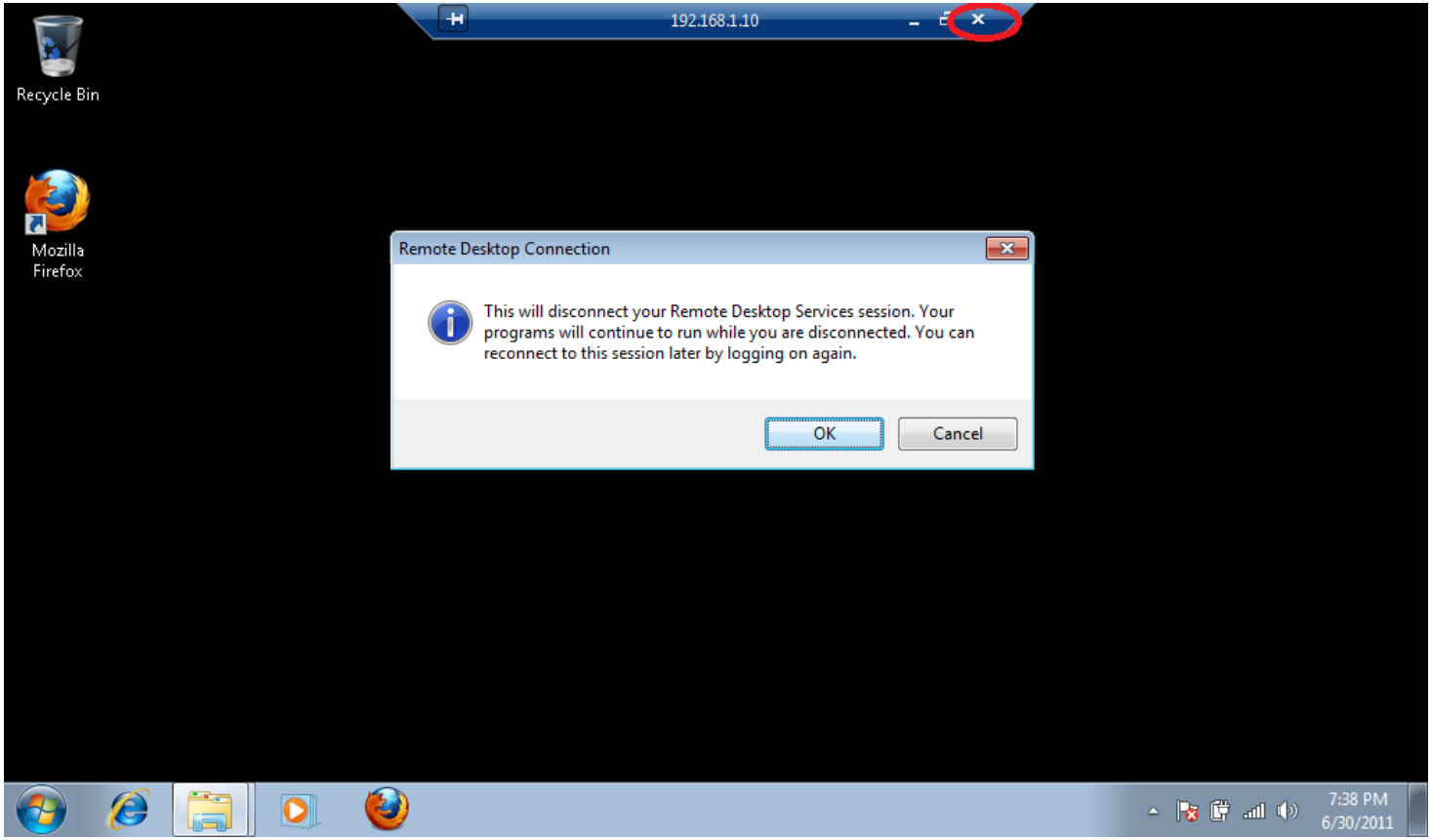
WCE: ‘Robar’ Credenciales en memoria

RDP ´expone´ credenciales: Ejemplo



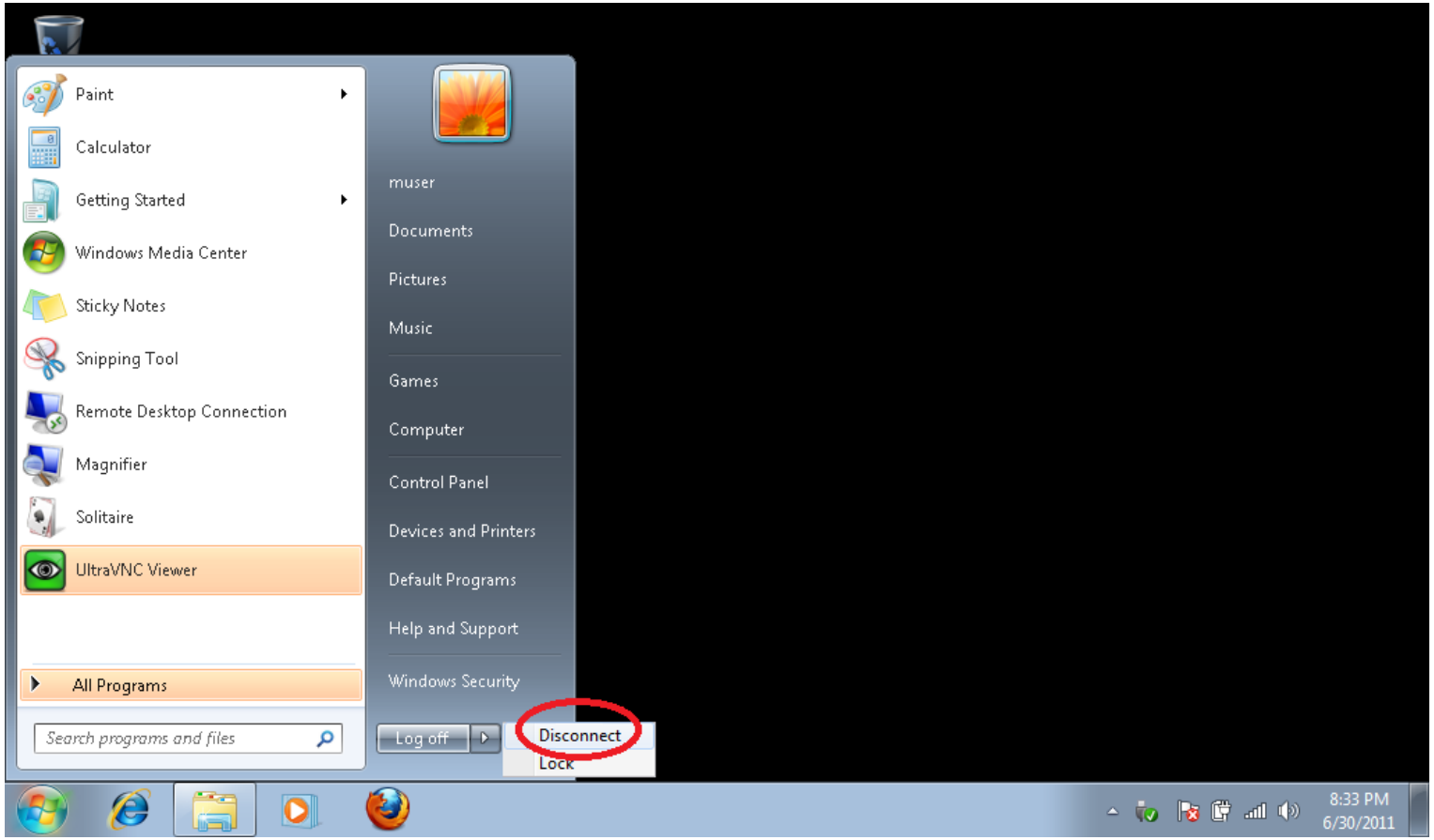
WCE: ‘Robar’ Credenciales en memoria

RDP: Desconectar != Log Off



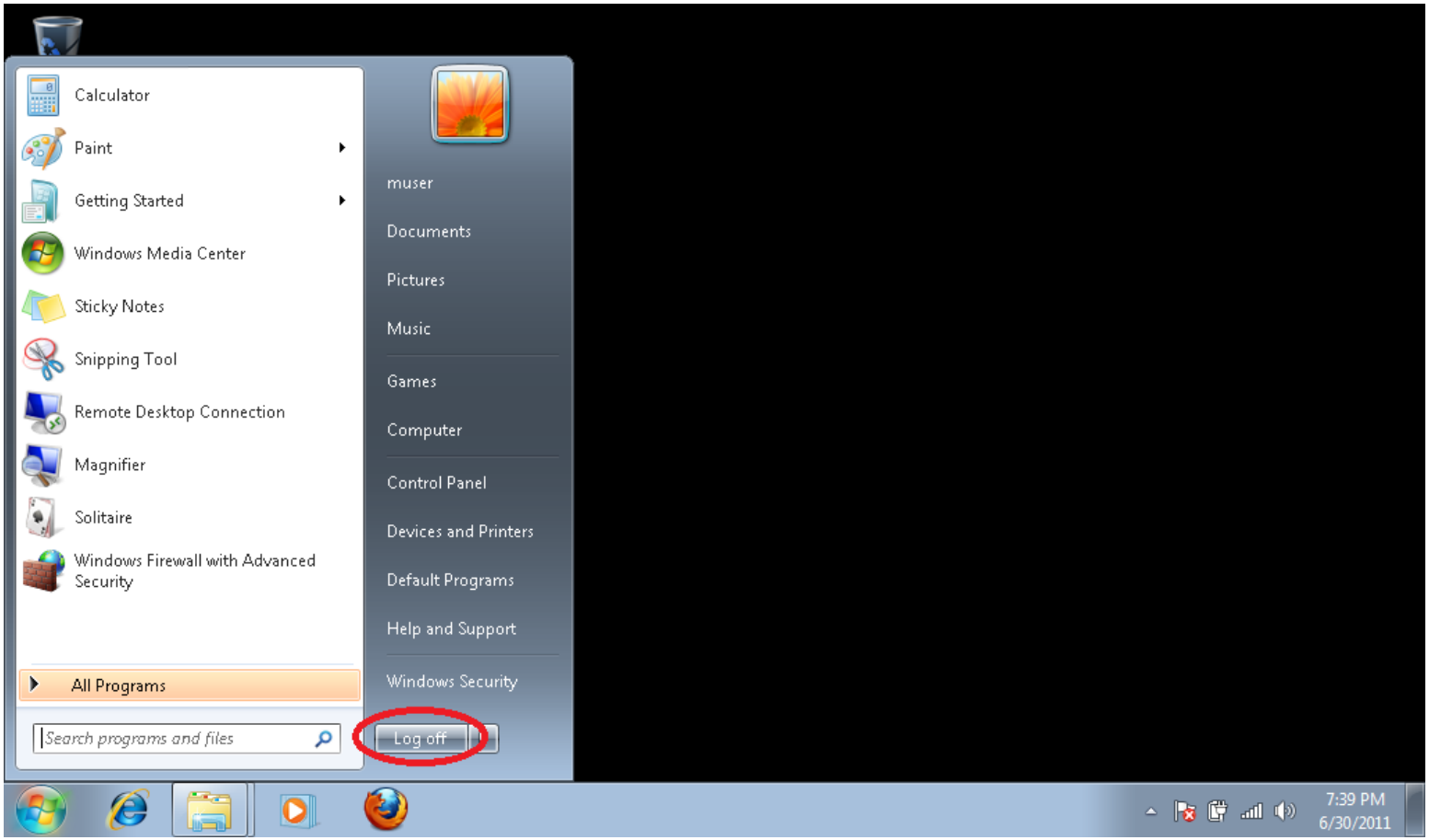
WCE: ‘Robar’ Credenciales en memoria

RDP: Desconectar != Log Off



WCE: ‘Robar’ Credenciales en memoria

RDP: Desconectar != Log Off



WCE: ‘Robar’ Credenciales en memoria

RDP: Desconectar != Log Off

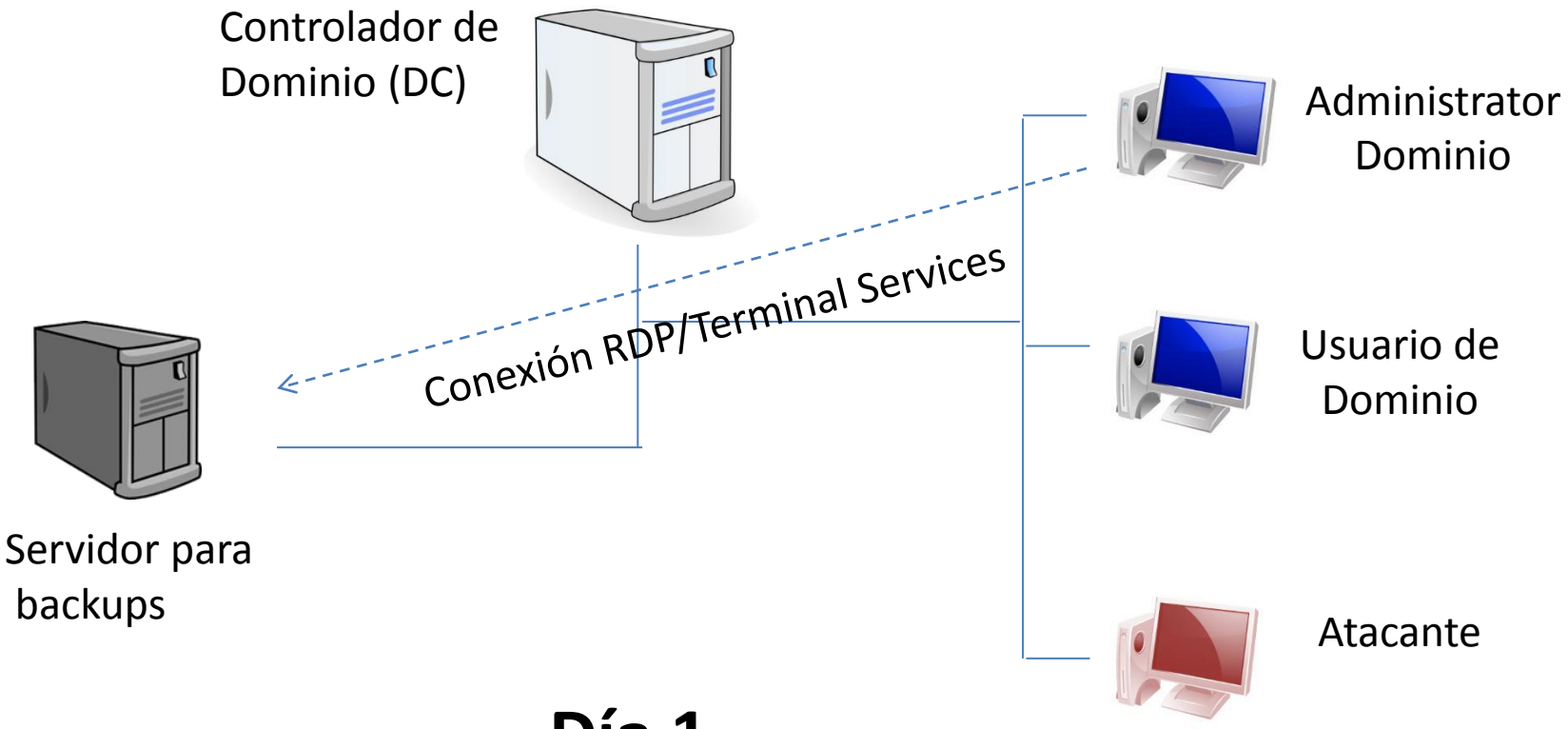
- **´Desconectar´ deja hashes NTLM en memoria**
  - La sesión de logon no es terminada
- ´Log Off´ termina la sesión de logon
  - Los hashes son borrados de memoria
- **Hay que hacer ´Log Off´!**
  - Es común que sólo se haga ´Desconectar´…
  - Inclusive los Administradores..

WCE: ‘Robar’ Credenciales en memoria

Sesiones de Logueo Zombies

WCE: ‘Robar’ Credenciales en memoria

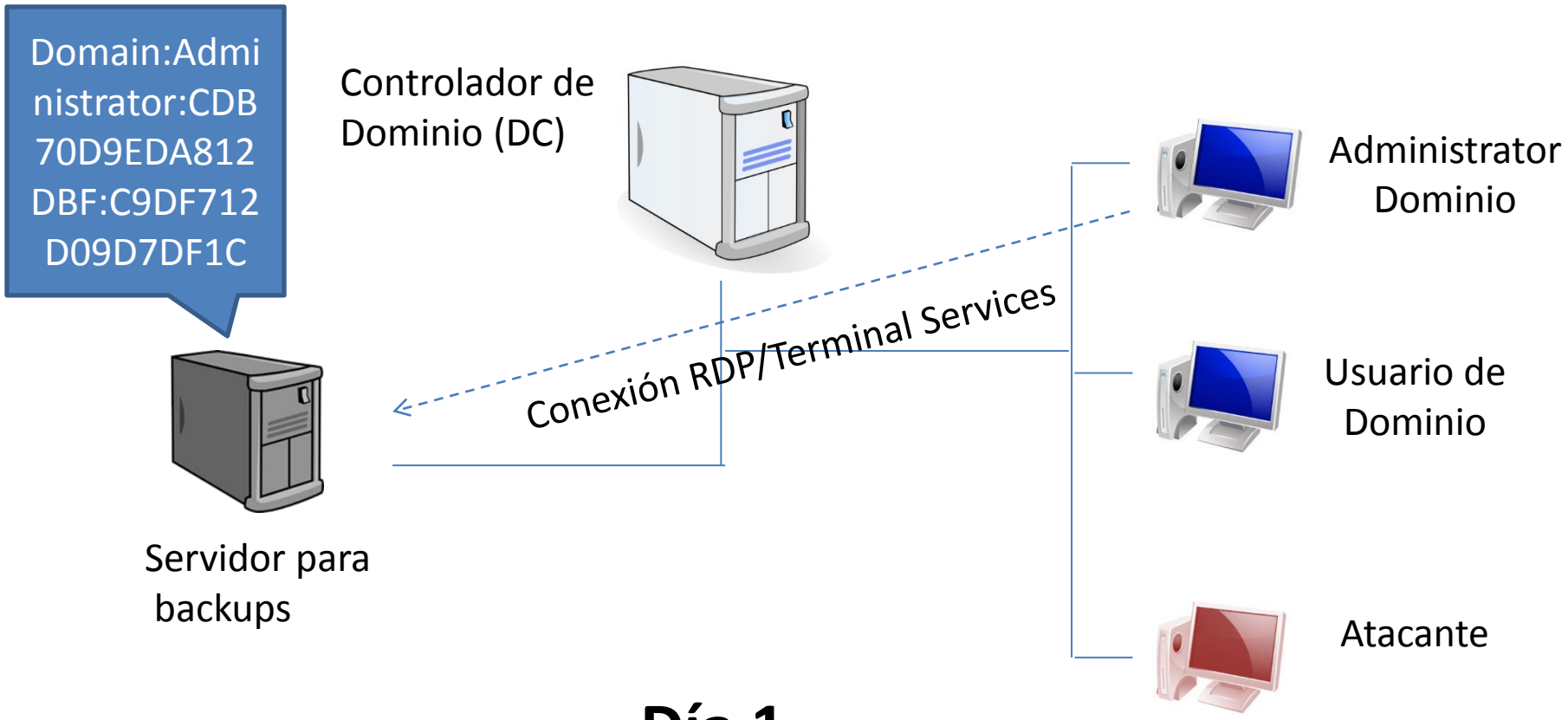
Bug: Sesiones de Logueo Zombies!



Día 1

WCE: ‘Robar’ Credenciales en memoria

Bug: Sesiones de Logueo Zombies!



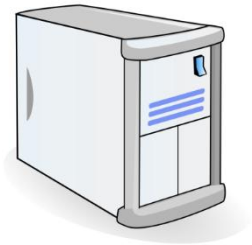
Día 1

WCE: ‘Robar’ Credenciales en memoria

Bug: Sesiones de Logueo Zombies!

Domain:Administrator:CDB70D9EDA812DBF:C9DF712D09D7DF1C

Controlador de Dominio (DC)



Administrator Dominio

Usuario de Dominio

Atacante

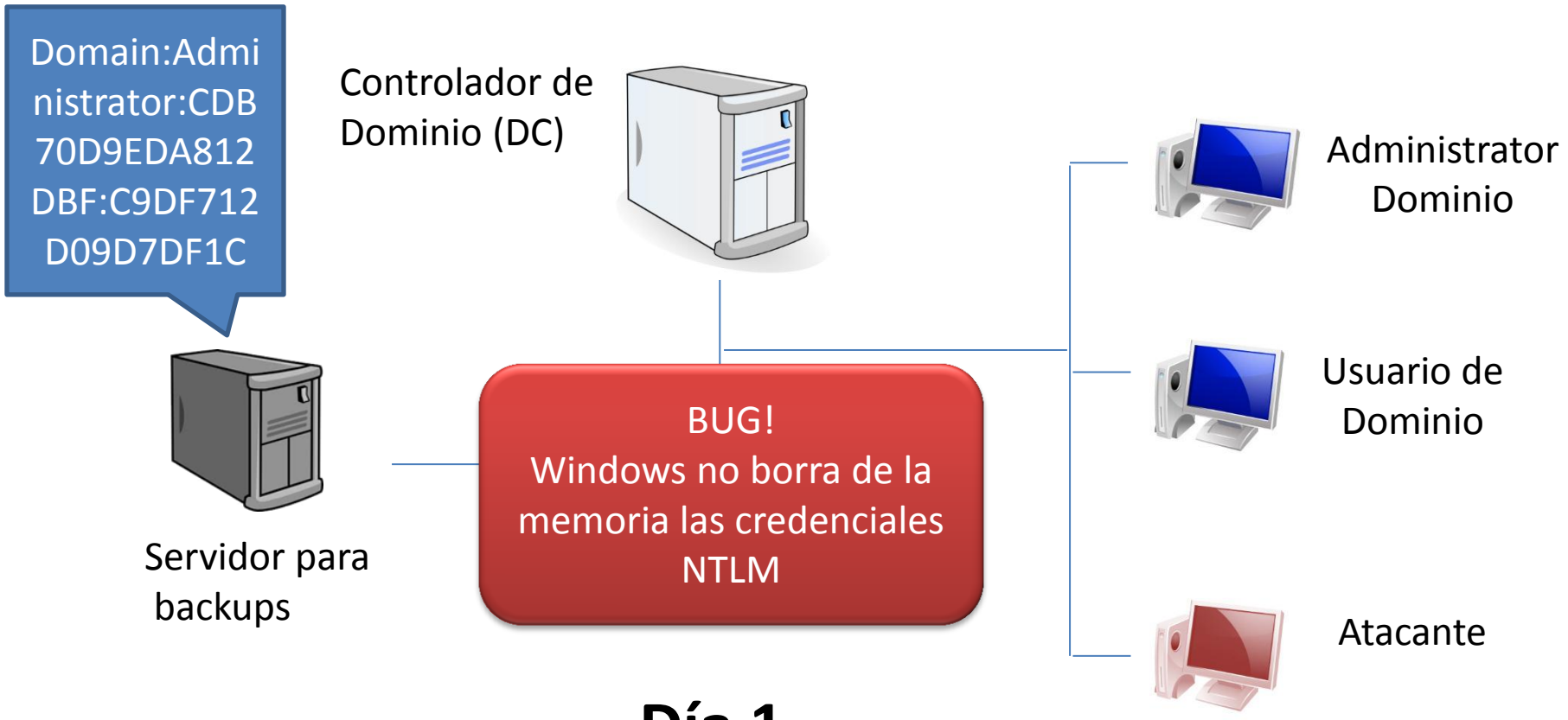
Servidor para backups

- El administrator hace log off, se desconecta, terminó de hacer su trabajo..

Día 1

WCE: ‘Robar’ Credenciales en memoria

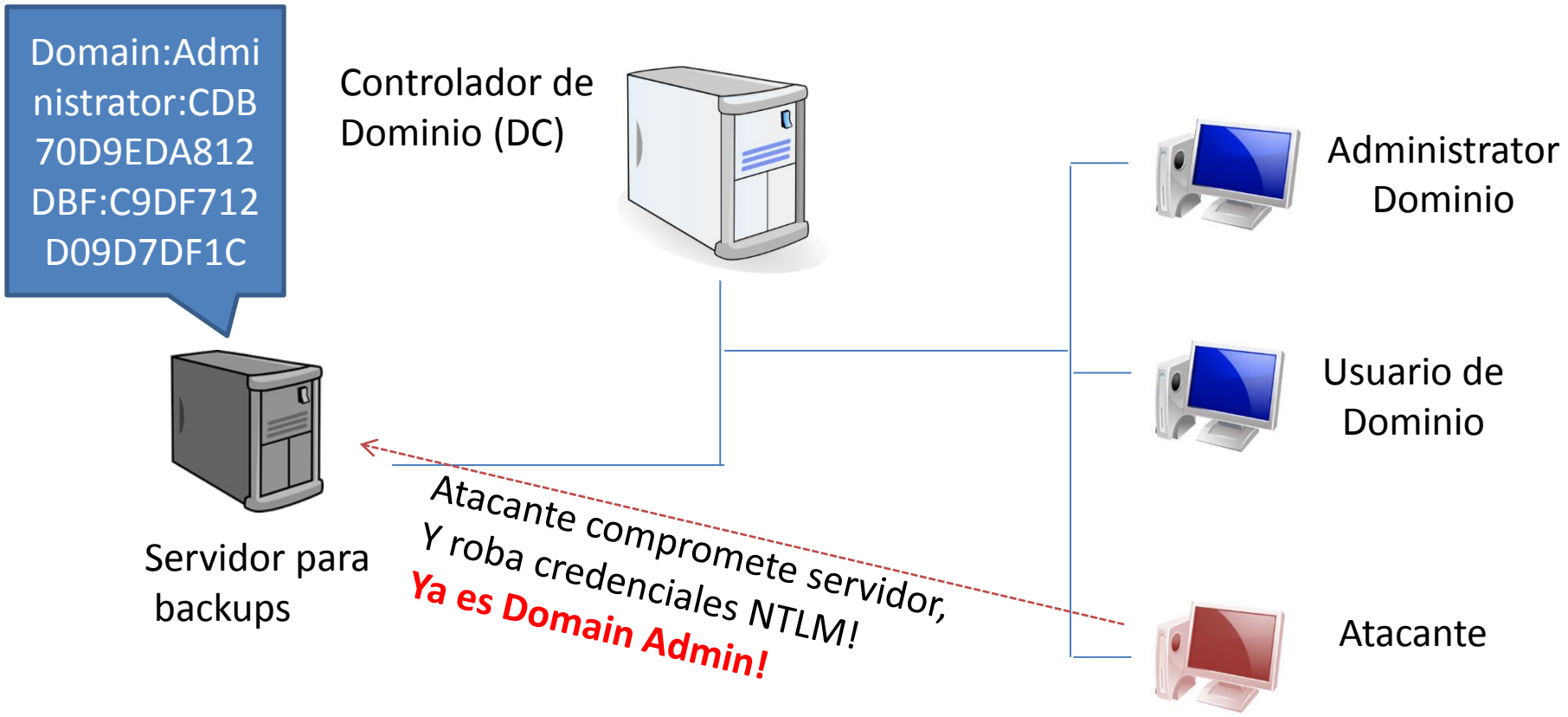
Bug: Sesiones de Logueo Zombies!



Día 1

WCE: ‘Robar’ Credenciales en memoria

Bug: Sesiones de Logueo Zombies!



Día 31.. ~ Un mes después..

WCE: Pass-the-Ticket (Kerberos)

- Nuevo ataque implementado en WCE v1.2
  - Primera y única herramienta que lo implementa AFAIK
- Post-explotación
- Equivalente a Pass-The-Hash para NTLM
- Permite 'robar' TGT/tickets Kerberos y usarlos en otros Windows y *Unix

WCE: Pass-the-Ticket (Kerberos)

- Tickets ‘robados’ son usados para acceder a servicios
  - Ejemplo: SMB para acceder a carpetas compartidas
- El TGT (Ticket Granting Ticket) permite crear tickets
  - Obtener acceso a otros servicios

Conclusiones

- WCE proporciona nuevas técnicas de post-explotación
  - Pass-the-Hash (NTLM)
  - Robar credenciales NTLM de memoria
  - Pass-the-ticket (Kerberos)
- útiles y usadas en pruebas de intrusión
- Necesario conocerlas para defenderse
  - No sólo para atacar..

Más información

- Presentación “WCE Internals”
  - RootedCon 2011; Madrid, España
  - Más detalles técnicos de implementación

http://www.ampliasecurity.com/research/WCE_Internals_RootedCon2011_ampliasecurity.pdf

Preguntas?

Gracias!

hernan@ampliasecurity.com

@hernano
@ampliasecurity

www.ampliasecurity.com