Detección de intrusiones empleando técnicas de cyber-deception con credenciales señuelo en Directorio Activo

Luis Ruiz Mayorga

Índice

Introducción y motivación

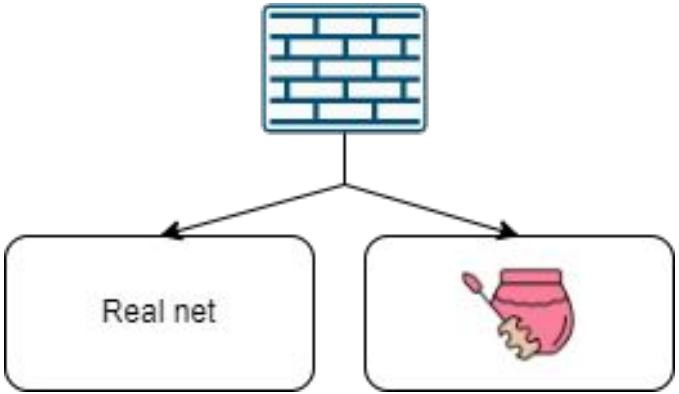
Introducción y motivación

- Incremento de la complejidad de los ataques de los cibercriminales.
- Necesidad de mejorar las herramientas de ciberdefensa para detectar intrusiones de la forma más temprana posible y minimizar el impacto
- Técnicas de deception
- Desarrollo de herramientas de código libre que implementen técnicas de deception

De las honeypots a deception

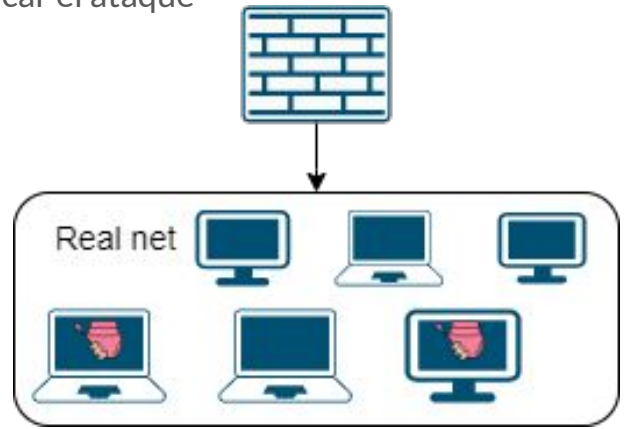
Honeypots

- Crear un entorno controlado que simula uno real.
- Desviar al atacante a este entorno falso.
- Evitar impacto a los entornos reales, estudiar comportamiento atacante



Deception

- Desplegar señuelos falsos en los entornos reales.
- Analizar cualquier interacción con los señuelos
- Identificar el ataque



Herramientas deception

Comerciales

- CounterCraft
- Rapid7
- Minerva Labs

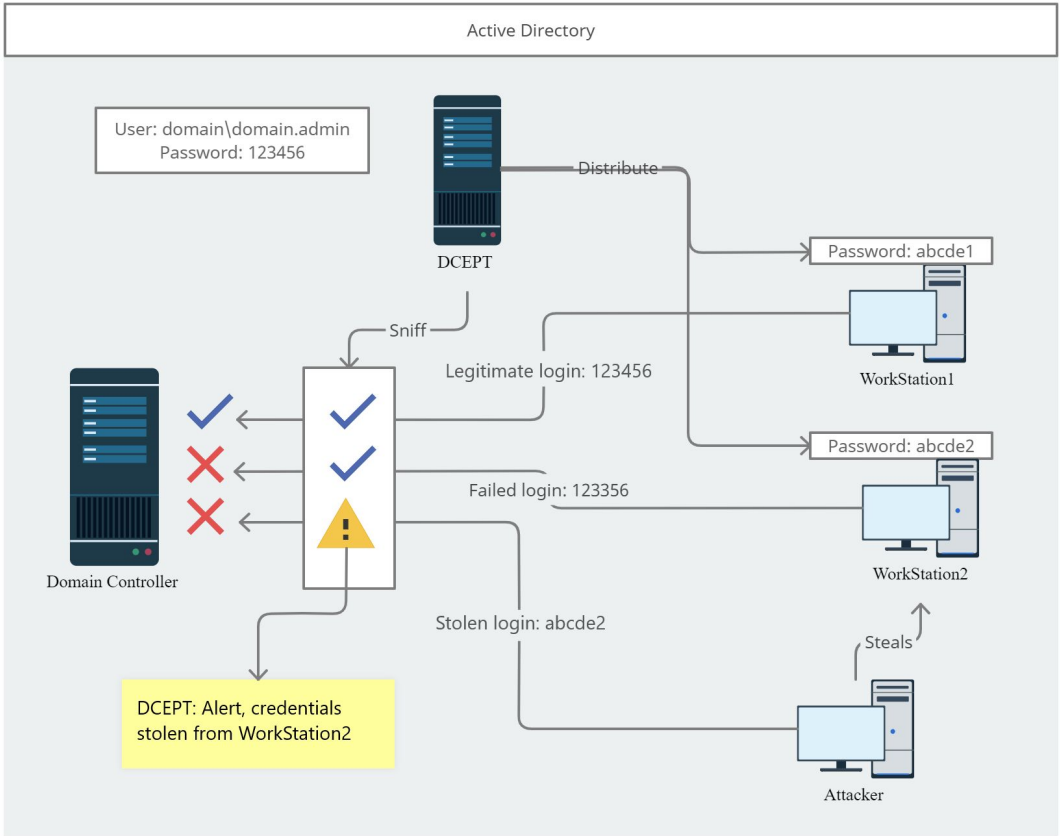
Código abierto

- Canarytokens
- Heralding
- Cowrie
- DejaVU
- T-Plot
- **DCEPT**

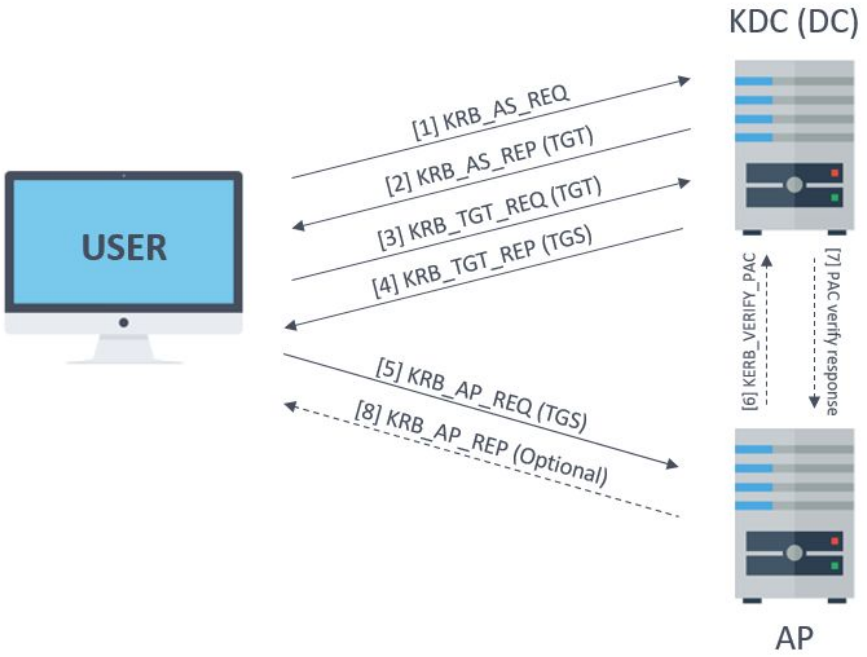
DCEPT es una prueba de concepto de **software libre sin alternativa** basada en el despliegue de **credenciales señuelo** en un entorno de **Kerberos**.

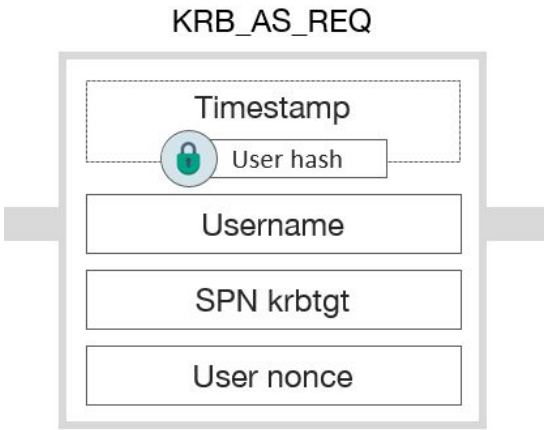
DCEPT

Funcionamiento



Kerberos


KDC (DC)
USER
[1] KRB_AS_REQ
[2] KRB_AS_REP (TGT)
[3] KRB_TGT_REQ (TGT)
[4] KRB_TGT_REP (TGS)
[6] KERB_VERIFY_PAC
[7] PAC verify response
[5] KRB_AP_REQ (TGS)
[8] KRB_AP_REP (Optional)
AP


KRB_AS_REQ
Timestamp
User hash
Username
SPN krbtgt
User nonce

Fortalezas y debilidades

Fortalezas

- No es necesario reconfigurar los Controladores de Dominio.
- No hace falta dar apariencia de usuarios reales a los señuelos, pues son los que ya existen en el dominio. No es posible que el atacante los identifique como señuelos
- Al utilizar credenciales falsas, no hay riesgo de que estos usuarios puedan ser utilizados en los equipos de forma maliciosa.
- En caso de detectarse el uso de unas credenciales robadas, puede identificarse el equipo en el que se ha producido.

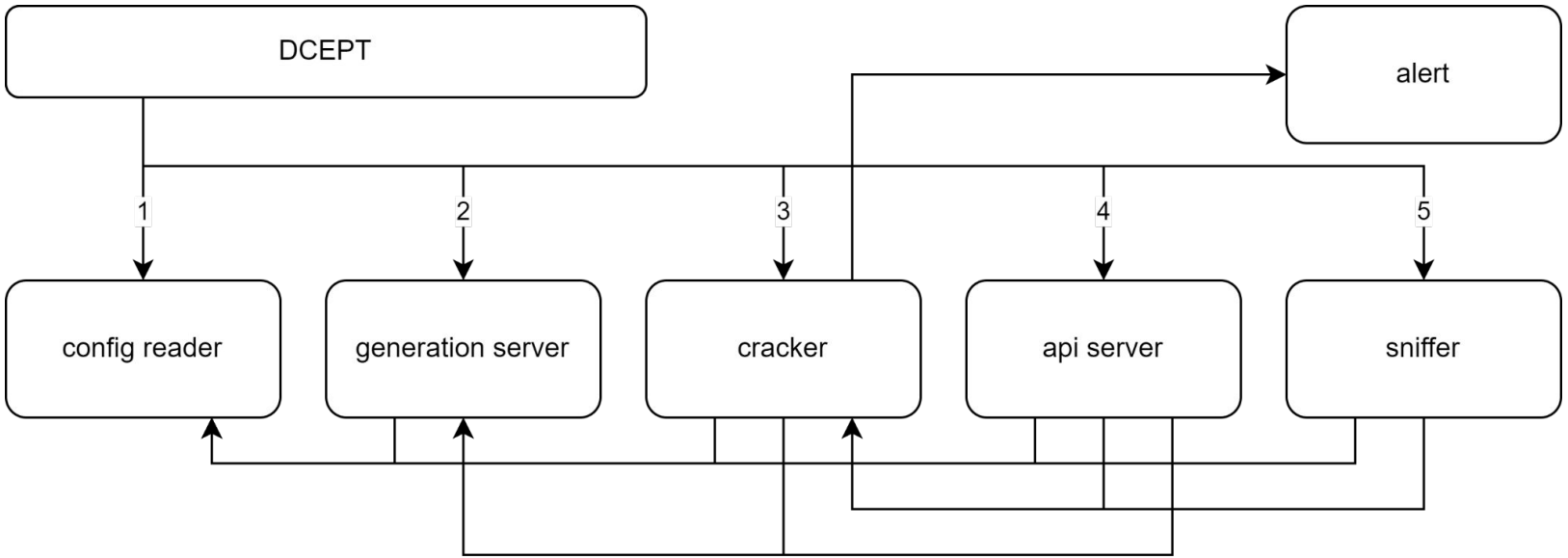
Debilidades

- Está programada en Python 2, que se encuentra fuera de soporte, y muchas distribuciones de Linux están dejando de lado esta versión, siendo complicado el despliegue en distribuciones actuales. Las dependencias han dejado de soportar esta versión de Python.
- Soporte de un bajo número de cifrados de Kerberos, pues la herramienta solo soporta los hashes con etype 18 sin sal.
- Algunas decisiones de diseño del código hacen complicado entender su funcionamiento y mantenerlo.
- El proyecto no ha recibido ningún tipo de soporte, actualización o mejora desde que se publicó en 2016.

Objetivo: corregir debilidades

Resultados y validación

Arquitectura DCEPT

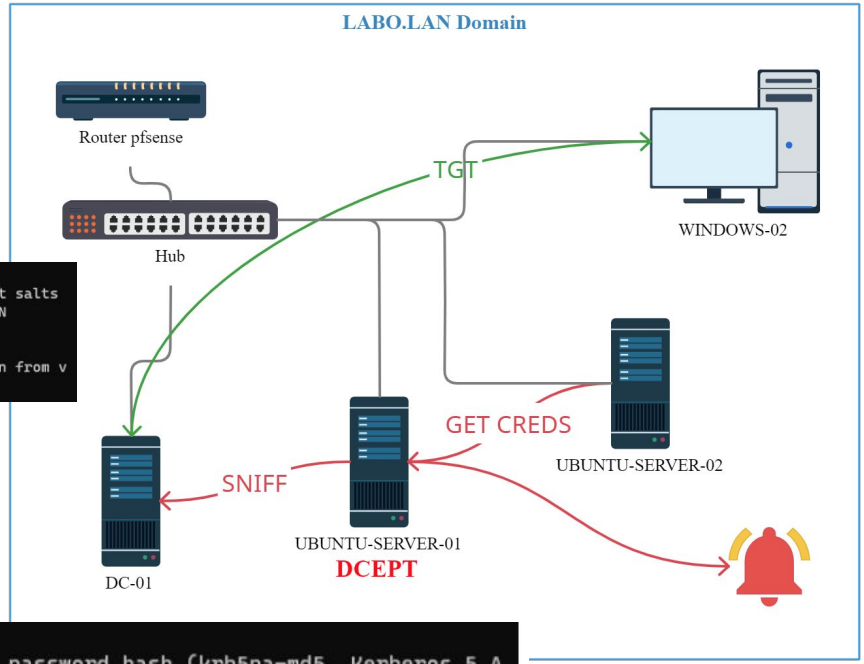


Definición de escenarios

Para definir los escenarios se ha tratado de escoger los más representativos para las capacidades más complejas que implementa DCEPT.

- Escenario 1: DCEPT como único nodo y cifrado actual.
- Escenario 2: DCEPT como único nodo y cifrado antiguo (etype 23).
- Escenario 3: DCEPT con dos nodos. Escenario en el que existan más de un controlador de dominio.

Escenarios 1 y 2



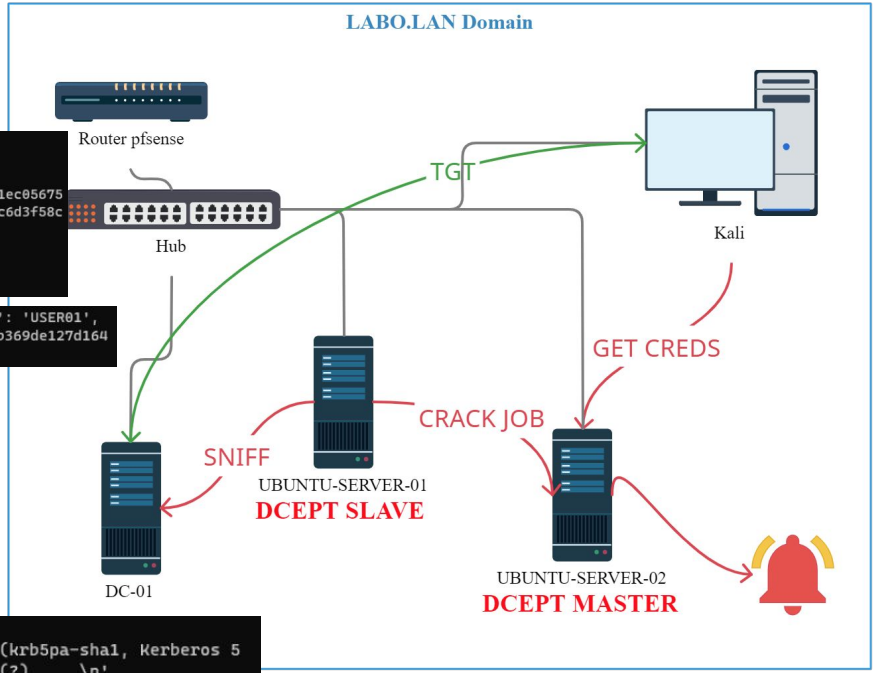

```
2022-04-24 14:29:39,301 DEBUG Cracking job completed
2022-04-24 14:29:39,301 DEBUG Cracking tool output: b'Loaded 2 password hashes with 2 different salts
(krb5pa-sha1, Kerberos 5 AS-REQ Pre-Auth etype 17/18 [PBKDF2-SHA1 128/128 SSE2 4x])\nWWGnDdDEpN
(?)     \n'
2022-04-24 14:29:39,302 INFO Cracked! Password: WWGnDdDEpN
2022-04-24 14:29:39,302 CRITICAL [ALERT] Honeytoken for LABO.LAN\user01 'WWGnDdDEpN' was stolen from v
alidation
```

```
2022-04-24 15:03:02,388 DEBUG Cracking job completed
2022-04-24 15:03:02,388 DEBUG Cracking tool output: b'Loaded 1 password hash (krb5pa-md5, Kerberos 5 A
S-REQ Pre-Auth etype 23 [32/64])\nFjc1EzBMIq       (?)     \n'
2022-04-24 15:03:02,388 INFO Cracked! Password: Fjc1EzBMIq
2022-04-24 15:03:02,389 CRITICAL [ALERT] Honeytoken for LABO.LAN\user01 'Fjc1EzBMIq' was stolen from v
alidate
```

Escenario 3

```
2022-04-24 15:48:04,253 DEBUG kerb-as-req for domain user LABO.LAN\USER01
2022-04-24 15:48:04,253 DEBUG No PA-DATA PA-ENC-TIMESTAMP in kerb-as-req for 'LABO.LAN\USER01'
2022-04-24 15:48:04,258 DEBUG kerb-as-req for domain user LABO.LAN\USER01
2022-04-24 15:48:04,259 INFO Ready to crack (user:USER01 domain:LABO.LAN etype:18 timestamp:861ec05675
766d0be80ad8b7bb369de127d164d4d2248a8a7b4829a5273785202f0208424f148fe4d8cee4caea9454c17d1a088dc6d3f58c
)
2022-04-24 15:48:04,259 DEBUG Sending cracking task to master node
2022-04-24 15:48:04,260 DEBUG Starting new HTTP connection (1): 192.168.10.21:8080
2022-04-24 15:48:04,283 DEBUG http://192.168.10.21:8080 "POST /notify HTTP/1.1" 200 3
```

```
2022-04-24 15:48:04,277 INFO Remote crack request recived from 192.168.10.22: {'kerb_name': 'USER01',
'kerb_realm': 'LABO.LAN', 'kerb_etype': '18', 'enc_timestamp': '861ec05675766d0be80ad8b7bb369de127d164
d4d2248a8a7b4829a5273785202f0208424f148fe4d8cee4caea9454c17d1a088dc6d3f58c'}
```




```
2022-04-24 15:51:23,038 DEBUG Cracking job completed
2022-04-24 15:51:23,039 DEBUG Cracking tool output: b'Loaded 1 password hash (krb5pa-sha1, Kerberos 5
AS-REQ Pre-Auth etype 17/18 [PBKDF2-SHA1 128/128 SSE2 4x])\ntestpassword     (?)     \n'
2022-04-24 15:51:23,039 INFO Cracked! Password: testpassword
2022-04-24 15:51:23,040 CRITICAL [ALERT] Honeytoken for LABO.LAN\user01 'testpassword' was stolen from
 TEST-PC
```

Conclusiones, líneas futuras y desafíos

Conclusiones y líneas futuras

Objetivos cumplidos

Está programada en Python 2, que se encuentra fuera de soporte, y muchas distribuciones de Linux están dejando de lado esta versión, siendo complicado el despliegue en distribuciones actuales. Las dependencias han dejado de soportar esta versión de Python.

Soporte de un bajo número de cifrados de Kerberos, pues la herramienta solo soporta los hashes con etype 18 sin sal.

Algunas decisiones de diseño del código hacen complicado entender su funcionamiento y mantenerlo.

El proyecto no ha recibido ningún tipo de soporte, actualización o mejora desde que se publicó en 2016.

Líneas futuras

- Mejor concurrencia entre hilos y la cola de tareas. Mayor coherencia al sistema de logging. Ampliar capacidades de gestión de errores.
- Permitir apagados y encendidos sin perder las tareas de descifrado pendientes.
- Implementar más sistemas de notificación: syslog, webhooks, etc.
- Soporte de forma simultánea de varios usuarios señuelo.
- Estudio y prueba de nuevos casos de uso de autenticación de Kerberos no cubiertos en este trabajo.
- Uso de GPU y Hascat para acelerar el proceso de descifrado de los paquetes de Kerberos.
- Utilizar protocolos seguros durante las comunicaciones, implementar las distintas APIs por HTTPS.

Principales desafíos del trabajo

- Trabajar con el código de otros desarrolladores
- Migración de Python 2 a Python 3
- Emular un entorno de Directorio Activo en el que ejecutar DCEPT
- Escasa información disponible de forma pública del funcionamiento detallado de Kerberos y su implementación en Directorio Activo

Preguntas