Fixing a Kerberos vulnerability with the bare necessities

Bronze-Bit exploit mitigation on old FreeIPA releases

Julien Rische
jrische@redhat.com
2024-02-04 FOSDEM

Red Hat France
About Kerberos

- Symmetric cryptography-based authentication protocol
- Created in 1988
- Early implementation of Single-Sign-On principles
- Use specific concepts
  - **Ticket**: Token used to authenticate a user or service against another service
  - **Key Distribution Center (KDC)**: Server storing all the keys and providing tickets to authenticated clients
  - **Ticket-Granting Ticket (TGT)**: Ticket to the KDC
The MS-SFU Kerberos extension
- Need to allow frontend services to impersonate users
  - Frontend: web service, ...
  - Backend: SQL database, distributed storage system, ...
- Historical solution: **TGT forwarding** (aka. *unconstrained delegation*)
  - Allow frontend service to access ANY service as the user
  - Bad solution from security perspective, more **granularity** required
- Microsoft implemented an extension called **MS-SFU**
  - Introducing 2 new mechanisms
Constrained Delegation (S4U2Proxy)

- Allow a **proxy** service to impersonate a **user** against a specific **target** service
- Configure service **delegation rules**
  - `ipa servicedelegation` commands
  - Specific administration permissions required to configure such rules
- At the condition of providing an **evicence ticket** to the **KDC**
  - Ticket for user-to-proxy service
  - With **forwarable** ticket flag set
Constrained Delegation (S4U2Proxy)
Constrained Delegation (S4U2Proxy)
Constrained Delegation (S4U2Proxy)
Constrained Delegation (S4U2Proxy)
Constrained Delegation (S4U2Proxy)

Diagram showing the flow of interactions between the user, proxy service, target service, and FreeIPA KDC. The process includes:
- User (U) requests a proxy (P) to perform an action.
- The proxy (P) requests a ticket (T) from the FreeIPA KDC.
- The target service (T) checks if the proxy is forwardable and delegatable.

Key steps:
- User → Proxy Service
- Proxy Service → Target Service
- Target Service → FreeIPA KDC
- FreeIPA KDC → Target Service

Additional elements:
- S4U2Proxy
- Check for forwardable and delegation rules
- Sname: Proxy
- Cname: User
- Forwardable 1
- ...
Constrained Delegation (S4U2Proxy)

User → Proxy Service → Target Service → FreeBSD KDC

S4U2Proxy

P → F

P → T

U → T

U → P

Check forwardable and delegation rules

Request completed

Access granted as User

sname: Proxy

cname: User
forwardable 1
...

5
Protocol Transition (S4U2Self)

- Mean to:
  - Integrate services relying different authentication methods for users requests into the Kerberos authentication system
    - OIDC, SASL, ...
  - Obtain encrypted user authorization information
    - Use Kerberos as group membership provider
- Allow **any service with a valid TGT** to request a ticket from **any user to the service itself**
- Resulting ticket has **forwardable** flag set only if:
  - FreeIPA: principal configured with `ok-to-auth-as-delegate` privilege
  - AD: account configured with `TrustedToAuthForDelegation` privilege
Protocol Transition (S4U2Self)

Proxy Service

FreelPA KDC

Proxy
ok-to-auth-as-delegate: false
Protocol Transition (S4U2Self)

Proxy Service

FreelPA KDC

Proxy
ok-to-auth-as-delegate: false

S4U2Self U

P \rightarrow F
Protocol Transition (S4U2Self)

- Proxy Service
- FreelPA KDC
  - Proxy
  - Key
  - Key
  - Key
  - Key
  - ok-to-auth-as-delegate: false

S4U2Self U

Set

forwardable

if Proxy has

ok-to-auth-as-delegate
Protocol Transition (S4U2Self)

- S4U2Self service
- Proxy service
- FreemIPA KDC
- Set `forwardable` if `Proxy` has `ok-to-auth-as-delegate`
Protocol Transition (S4U2Self)

Set
forwardable
if Proxy has
ok-to-auth-as-delegate

sname: Proxy

cname: User

Forwardable 0

...
Protocol Transition (S4U2Self)

Proxy Service

FREEIPA KDC

S4U2Self U

Set:
forwardable
if Proxy has
ok-to-auth-as-delegate

Proxy
ok-to-auth-as-delegate: true

sname: Proxy

cname: User

Forwardable

...


Protocol Transition (S4U2Self)

- **Proxy Service**
- **FreelPA KDC**
- **S4U2Self U**

Set `forwardable` if **Proxy** has `ok-to-auth-as-delegate`

- **Proxy**
  - `ok-to-auth-as-delegate: true`

- **User**
  - `cname: User`
  - `forwardable 1`
  - `...`
Protocol Transition (S4U2Self)

- **Proxy Service**
- **FreelPA KDC**
- **S4U2Self U**:
  - \( P \rightarrow F \)

**Proxy**
- `ok-to-auth-as-delegate: true`

- **Set**
  - `forwardable`
  - If **Proxy** has `ok-to-auth-as-delegate`

- **sname:** Proxy
  - **cname:** User
    - `Forwardable 1`
    - `...`
Protocol Transition (S4U2Self)

Proxy Service

FreelPA KDC

Proxy
ok-to-auth-as-delegate: true

S4U2Self U

P → F

U → P

Set
forwardable
if Proxy has
ok-to-auth-as-delegate

sname: Proxy

cname: User

Forwardable 1

...

The Bronze-Bit exploit
The problem with MS-SFU

- A service with the `forwardable` S4U2Self ticket permission AND a constrained delegation rule can impersonate any user against the target service of this delegation rule
  - Including users with administration privileges for this service
- The `forwardable` flag is encrypted using the proxy service key
  - But nothing keeps the service from changing the value of this flag
- If the host running the proxy service is compromised, the attacker could use proxy service’s credentials to access the target service as an admin user
The Bronze-Bit exploit
The Bronze-Bit exploit
The Bronze-Bit exploit
The Bronze-Bit exploit

Proxy Service

Target Service

FreeIPA KDC

S4U2Self U

Proxy does not have ax-to-auth-as-delegate. Hence forwardable is not set

Decode encrypted part, flip the forwardable flag, and re-encrypt

sname: Proxy

cname: User

forwardable: 0

...
The Bronze-Bit exploit
The Bronze-Bit exploit

Decide encrypted part, flip the forwardable flag, and re-encrypt

Proxy does not have set to-auth-as delegate, hence forwardable is not set
The Bronze-Bit exploit
The Bronze-Bit exploit

Proxy Service

Target Service

FreeIPA KDC

S4U2Self U

P → F

U → P

S4U2Proxy T

P → F

U → P

Proxy does not have "kt-to-svr-as-delegate" hence Forwardable is not set

Forwardable is set and delegation rule exists

sname: Proxy

cname: User

forwardable: 1

...

Decode encrypted part, flip the forwardable flag, and re-encrypt
The Bronze-Bit exploit

- **Proxy Service**
  - `U → P`
  - Decode encrypted part, flip the `forwardable` flag, and re-encrypt

- **Target Service**
  - `P → F`
  - `S4U2Self U`

- **FreeIPA KDC**
  - `F`

- **Proxy**
  - `P → F`
  - `Proxy` does not have `x64-to-wink-as-delegate`, hence `forwardable` is not set

- **Forwardable**
  - `forwardable`
  - `forwardable: 1`
  - `cname: User`
  - `...`
The Bronze-Bit exploit

Decoded encrypted part, flip the forwardable flag, and re-encrypt

Forwardable is set and delegation rule exists

sname: Proxy

cname: User

forwardable: 1

...
Reproducer for MIT Kerberos and FreeIPA

- All available reproducers designed for Active Directory
- None of them could work against FreeIPA, because they were missing support for:
  - `PA_S4U_X509_USER` ASN.1 sequence (for S4U2Proxy)
  - AES HMAC-SHA2 encryption types family (from RFC8009)
- We implemented support for these 2 features in the **Impacket Python library**
  - `fortra/impacket#1684`:
    - Implement Kerberos encryption types from RFC8009 (AES HMAC-SHA2 family)
    - [https://github.com/fortra/impacket/pull/1684](https://github.com/fortra/impacket/pull/1684)
Fix: Ticket signature

- Solution designed by Microsoft
  - **Signature** actually means **keyed checksum** (RFC3961, RFC4120)
- Implemented by AD and MIT Kerberos 1.20
- Sign the encrypted part of the ticket using the **KDC key**
  - KDC able to detect any modification of ticket’s encrypted part
  - *forwardable* flag protected
- MS-PAC Kerberos extension
  - Add a **Privilege Attribute Certificate** (PAC) in the ticket
PAC ticket signature
PAC ticket signature

Proxy Service

Target Service

FreeIPA KDC

S4U2Self U

Proxy does not have S2T, or w/ a delegate, hence forwardable is not set

pname: Proxy

cname: User

Forwardable 0

...

Authorization Data

...

PAC

...

12
PAC ticket signature

Proxy Service

Target Service

FreelPA KDC

S4U2Self U

P → F

Proxy does not have
S4U to Target delegate
hence forwardable
is not set

sname: Proxy

cname: User

Forwardable 0

...

Authorization Data

...

PAC

...

Ticket signature
PAC ticket signature

Proxy Service | Target Service | FreeIPA KDC

S4U2Self U

P → F

Proxy does not have S4U rights to delegate, hence forwardable is not set

U → P

sname: Proxy

cname: User

Forwardable 0

...

Authorization Data

...

PAC

...

Ticket signature
PAC ticket signature

Proxy Service

Target Service

FreelPA KDC

Decoded encrypted part, flip the Forwardable flag, and re-encrypt

Proxy does not have S4U2Self U permission, hence Forwardable is not set

Proxy does not have S4U2Self U permission, hence Forwardable is not set

Ticket signature
PAC ticket signature

Proxy Service → S4U2Self U → P

Target Service

FreeIPA KDC

Proxy does not have "forwardable" set, hence "delegate" is not set

Decode encrypted part, flip the "forwardable" flag, and re-encrypt

sname: Proxy

cname: User

Forwardable 0

...

Authorization Data

...

PAC

...

Ticket signature
PAC ticket signature

- Proxy Service
- Target Service
- FreeIPA KDC

S4U2Self U

P → F

Proxy does not have S4U with a delegate, hence Forwardable is not set.

Decode encrypted part, flip the Forwardable flag, and re-encrypt.

U → P

Proxy

cname: User
Forwardable 1
...
Authorization Data
...
PAC
...
Ticket signature
PAC ticket signature

- Proxy
- Target Service
- FreeIPA KDC

S4U2Self U
\[ P \rightarrow F \]
\[ U \rightarrow P \]

S4U2Proxy T
\[ P \rightarrow F \]
\[ U \rightarrow P \]

Proxy does not have proxy-with-delegate flag, hence forwardable is not set.

Decode encrypted part, flip the forwardable flag, and re-encrypt.

sname: Proxy

cname: User
Forwardable 1

Authorization Data

... PAC

Ticket signature
PAC ticket signature

- Proxy Service
- Target Service
- FreeIPA KDC

- S4U2Self U: $P \rightarrow F$
- $U \rightarrow P$

- S4U2Proxy T: $P \rightarrow F$
- $U \rightarrow P$

- Proxy does not have $S$ to auth as delegate, hence $Forwardable$ is not set

- Check PAC ticket signature, if valid [...]

- Ticket signature

- sname: Proxy
  - cname: User
    - Forwardable 1
    - Authorization Data
      - ...
      - PAC
        - ...
        - Ticket signature
PAC ticket signature

Proxy Service

Target Service

FreeIPA KDC

S4U2Self U

S4U2Proxy T

Decode encrypted part, flip the forwardable flag, and re-encrypt

PAC ticket signature

Check PAC ticket signature, if valid [...]
Fix for CentOS 8 Stream and RHEL 8
C8S/RHEL8: Software constraints

- Using MIT Kerberos 1.18
- PAC generation handled by IPA KDB plugin
- ABI compatibility within major release
  - Update to MIT krb5 1.20 impossible
- PAC ticket signature not backportable

```c
krb5_error_code
(*sign_authdata)(krb5_context kcontext,
    krb5_const_principal client_princ,
    krb5_db_entry *client,
    krb5_db_entry *header_server,
    krb5_keyblock *client_key,
    krb5_keyblock *header_key,
    krb5_keyblock *session_key,
    krb5_authdata **tgt_auth_data,
    krb5_data ***auth_indicators,
    krb5_keyblock *session_key,
    krb5_keyblock *server_key,
    krb5_DB *header_db,
    krb5_DB *local_db,
    krb5_db_entry *header_entry,
    krb5_db_entry *local_entry,
    void *ad_info,
    krb5_keyblock **tgt_keyblock,
    krb5_error_code **error)
```
• If the ticket cannot be protected, maybe the KDC could detect the attack.

• The PAC contains **additional authorization information**
  - List of SIDs

• **Security identifier (SID)**
  - Identifiers used in the AD world
  - Unique, except for some well-known ones

• Well-known SIDs supported by FreeIPA:
  - **S-1-18-1**: Authentication authority asserted identity
    - Ticket obtained using normal user request
  - **S-1-18-2**: Service asserted identity
    - Ticket obtained using S4U2Self
Bronze-Bit attack detection

Proxy Service  Target Service  FreeIPA KDC

S4U2Self U  P \rightarrow F

Proxy does not have ok-to-auth-as-delegate: false, hence Forwardable is not set

name: Proxy

cname: User
Forwardable 0
...

Authorization Data
...

PAC
Logon Info
...
S-1-18-2
...

15
Bronze-Bit attack detection

Proxy Service
Target Service
FreelPA KDC

S4U2Self U

P ← F

U ← P

Proxy does not have
ok-to-auth-as-delegate: false

Forwardable

Authorization Data

PAC
Logon Info
...
S-1-18-2
...

Decode encrypted part, flip the forwardable flag, and re-encrypt
Bronze-Bit attack detection

[Diagram showing S4U2Self and S4U2Proxy flows with Proxy Service, Target Service, and FreelPA KDC involved. The diagram explains how proxy does not have proxy-ok-to-auth-as-delegate, hence Forwardable flag is not set.]

S4U2Self U

P → F

U → P

S4U2Proxy T

P → F

U → P

Proxy does not have proxy-ok-to-auth-as-delegate, hence Forwardable flag is not set.

Decode encrypted part, flip the Forwardable flag, and re-encrypt.

PAC

Logon Info

S-1-18-2

...
Bronze-Bit attack detection

- Proxy Service
- Target Service
- FreeIPA KDC

Diagram:
- S4U2Self U
  - P → F
  - U → P
- S4U2Proxy T
  - P → F
  - U → P
- Proxy
  - ok-to-auth-as-delegate: false

Decoding:
- Decode encrypted part, flip the forwardable flag, and re-encrypt

Crosscheck:
- Ticket's forwardable flag, Proxy's ok-to-auth-as-delegate attribute, and PAC SIDs

Objects:
- sname: Proxy
- cname: User
  - Forwardable
  - Authorization Data

PAC:
- Logon Info
  - ...
- S-1-18-2
  - ...
- ...

States:
- Forwardable
- Authorization Data
- PAC

15
Bronze-Bit attack detection

Proxy Service
Target Service
FreeIPA KDC

S4U2Self U
\(P \rightarrow F\)
\(U \rightarrow P\)

S4U2Proxy T
\(P \rightarrow F\)
\(U \rightarrow P\)

Proxy does not have \(\text{ok-to-auth-as-delegate}\) set, hence \(\text{Forwardable}\) is not set

Crosscheck ticket's \(\text{Forwardable}\) flag, Proxy's \(\text{ok-to-auth-as-delegate}\) attribute, and PAC SIDs

PAC
Logon Info
\(\ldots\)
\(S-1-18-2\)
\(\ldots\)

sname: Proxy

cname: User

Forwardable1

Authorization Data
\(\ldots\)

Decode encrypted part, flip the \(\text{Forwardable}\) flag, and re-encrypt
Bronze-Bit attack detection
Bronze-Bit attack detection

Proxy Service
Target Service
FreelIPA KDC

S4U2Self U

Proxy
on-to-auth-as-delegate: false

S4U2Proxy T

Proxy does not have on-to-auth-as-delegate, hence Forwardable is not set

Crosscheck ticket's Forwardable flag, Proxy's on-to-auth-as-delegate attribute, and PAC SIDs

Decode encrypted part, flip the forwardable flag, and re-encrypt

Pname: Proxy

cname: User
Forwardable: 1
...
Authorization Data
...

PAC
Logon Info
...
S-1-18-1
...


CVE-2022-37967

- PAC spoofing
  - Authorization information can be modified
- MS-PAC updated to add the **extended KDC signature**
Bronze-Bit attack detection with PAC extended KDC signature

Proxy Service

Target Service

FreeIPA KDC

S4U2Self U

Proxy does not have ok-to-auth-as-delegate; hence forwardable is not set

Proxy

cname: Proxy

forwardable 0

...

Authorization Data

PAC

Logon Info

...

S-1-18-2

...


Bronze-Bit attack detection with PAC extended KDC signature
Bronze-Bit attack detection with PAC extended KDC signature
Bronze-Bit attack detection with PAC extended KDC signature

S4U2Self U

Proxy does not have ok-to-auth-as-delegate, hence forwardable is not set

Decode encrypted part, flip the forwardable flag, and re-encrypt

Proxy

Service

Target Service

FreiPA KDC

Proxy: ok-to-auth-as-delegate: false

sname: Proxy

cname: User

Forwardable 1

Authorization Data

PAC

Logon Info

... 

S-1-18-2

... 

Extended KDC sign.
Bronze-Bit attack detection with PAC extended KDC signature
Bronze-Bit attack detection with PAC extended KDC signature

- **Proxy Service**
- **Target Service**
- **FreeIPA KDC**

**S4U2Self U**

- **P → F**
- **U → P**

**Proxy**

- **does not have**
- **ok-to-auth-as-delegate**
- **hence is not set**

**S4U2Proxy T**

- **P → F**
- **U → P**

- **Decode encrypted part, flip the forwardable flag, and re-encrypt**

- **Verify extended KDC signature, crosscheck ticket's forwardable flag, Proxy's ok-to-auth-as-delegate attribute, and PAC SIDs**

- **Proxy KDC signature**
- **Forwardable**
- **Authorization Data**
- **Extended KDC sign.**
Bronze-Bit attack detection with PAC extended KDC signature

Proxy Service

Target Service

FreeIPA KDC

Proxy
ok-to-auth-as-delegate: false

S4U2Self U
P → F

U → P

Proxy does not have
ok-to-auth-as-delegate
hence forwardable
is not set

S4U2Proxy T
P → F

U → P

Verify extended KDC
signature, crosscheck ticket’s
forwardable flag, Proxy’s
ok-to-auth-as-delegate
attribute, and PAC SIDs

PAC
Logon Info
...
S-1-18-2
...

Extended KDC sign.

cname: Proxy

cname: User

Forwardable 1
...

Authorization Data
...

17
Bronze-Bit attack detection with PAC extended KDC signature

- Proxy Service
- Target Service
- FreeIPA KDC

1. S4U2Self U
   - P → F

2. Proxy does not have ok-to-auth-as-delegate: false hence forwardable is not set

3. S4U2Proxy T
   - P → F
   - U → P

4. Verify extended KDC signature, crosscheck ticket's forwardable flag, Proxy's ok-to-auth-as-delegate attribute, and PAC SIDs

5. sname: Proxy
   - cname: User
   - Forwardable 1
   - Authorization Data
   - PAC
     - Logon Info
     - ... S-1-18-2
     - ...
     - Extended KDC sign.
Bronze-Bit attack detection with PAC extended KDC signature
Conclusion
Conclusion

- Good example of the typical tribulations of long-term support
  - Especially for security-related network protocols
- MS-SFU is the continuation of Kerberos’ gradual shift
  - From authentication only to authentication and authorization
References

1. MS-SFU: Service for User and Constrained Delegation Protocol
2. FreeIPA General Constrained Delegation
   https://freeipa.readthedocs.io/en/ipa-4-10/designs/rbcd.html#general-constrained-delegation-design
3. [Blog] Kerberos: How does delegation work? (Tarlogic)
   https://www.tarlogic.com/blog/kerberos-iii-how-does-delegation-work/
4. [Blog] Kerberos constrained delegation with protocol transition (Phackt)
5. [Blog] Kerberos Delegation (Hackndo)
   https://en.hackndo.com/constrained-unconstrained-delegation/
7. KDC-REQ-BODY signature (RFC4120)
   https://datatracker.ietf.org/doc/html/rfc4120#section-5.2.7.1
8. RFC8009: AES Encryption with HMAC-SHA2 for Kerberos 5
9. impacket#1684: Implement Kerberos encryption types from RFC8009 (AES HMAC-SHA2 family)
   https://github.com/fortra/impacket/pull/1684
10. MS-PAC: ticket signature
    https://learn.microsoft.com/en-us/openspecs/windows_protocols/ms-pac/76c10ef5-de76-44bf-b208-0d8750fc2edd
11. Microsoft KB4598347 update
12. MIT Kerberos upstream pull request for PAC ticket signature
    https://github.com/krb5/krb5/pull/11225
13. RHEL8 Compatibility Levels
    https://access.redhat.com/articles/rhel8-abi-compatibility
14. MIT Kerberos 1.18.2 KDB plugin API
    https://github.com/krb5/krb5/blob/krb5-1.18.2-final/src/include/krb5/kdcpolicy_plugin.h#L1120-L1126
15. AD special identity groups
16. Service Asserted Identity SID set by FreeIPA for S4U2Self
    https://github.com/freeipa/freeipa/blob/release-4-9-12/demo/krb5kdc/krb5kdc/krb5kdc-1.18.2-final/src/include/krb5/kdcpolicy_plugin.h#L1120-L1126
17. Kerberos’ RC4-HMAC broken in practice: spoofing PACs with MD5 collisions
18. MS-PAC: extended KDC signature
    https://learn.microsoft.com/en-us/openspecs/windows_protocols/ms-pac/9cf6f6ad-6b7e-44ed-901aa1ff5a08
19. MIT Kerberos upstream pull request for PAC extended KDC signature
    https://github.com/krb5/krb5/pull/11284
20. Bronze-Bit attack detection for FreeIPA
    https://github.com/freeipa/freeipa/commit/a847e2483b4c832ee5129901da169f4eb0d1392
21. Build conditions for Bronze-Bit attack detection in FreeIPA
    https://github.com/freeipa/commit/6ca47ba092811029eed0f12f8af9c34ba7662924
22. Bronze-Bit attack detection patch for CentOS 8 Stream
    https://gitlab.com/redhat/centos-stream/rpms/ipa/-/merge_requests/58/
Questions?
Thank you!