MIT Kerberos and Cross Platform Interoperability with Microsoft and Java

Jeffrey Eric Altman

jaltman * secure-endpoints * com

http://www.secure-endpoints.com

Who am I and what do I do?

- MIT Kerberos for Windows maintainer
- OpenAFS Gatekeeper for Windows
- Internet Engineering Task Force (IETF)
- Project JXTA Board Member

secureendpoints

What else have I done?

- The Kermit Project
  - Cross platform (Unix, OS/2, Windows)
- Internet Access Methods
  - Java based Person to Person collaboration software
- Miscellaneous Network Security stuff
  - OpenSSL, Secure Remote Password, TELNET START_TLS, FTP AUTH TLS, SSH

secureendpoints

What is Secure Endpoints Inc.?

Secure Endpoints Inc. is a newly formed corporation dedicated to the support and development of open source security related communications technologies on the Microsoft Windows platform.

An overview of Today's talk

- What is Kerberos?
- What is GSS-API?
- Overview of MIT's Kerberos products
- Some basic Kerberos Configuration
- Interoperability Issues with Windows SSP Kerberos 5 and Java GSS
- Demos and Code Walkthrough

secureendpoints

What is Kerberos?

- Trusted Third Party Authentication Protocol
- Mutual Authentication and Key Exchange
- Symmetric Shared Key Cryptography
- Single Sign-On to multiple systems
- Cross-realm authentication for distributed management
- IETF Proposed Standard (KRB-WG)

secureendpoints

Operating Environments Shipping with Kerberos

- Microsoft Windows 2000, XP, 2003
- Mac OS X
- Linux
- Solaris
- AIX (DCE)
- Java
- others

How does it work?

- A trusted third party (the Key Distribution Center) shares a key (the secret) with all clients and services
- When Client A wants to authenticate with Service B, Client A must first prove its identity to the KDC which will provide a Ticket Granting Ticket to Client A.

secureendpoints

What is a Ticket Granting Ticket?

- A TGT is a ticket issued to a Client which can be used to request Service Tickets from a Ticket Granting Service
- The TGT provides Client A with a secret session key which can be used to encrypt messages sent between the Client and the TGS.
- The TGT includes an envelope encrypted with the TGS secret key known to KDC. This envelope includes a copy of the session key, Client A's name, and a window of validity

secureendpoints

Ticket Granting Service

- Client A requests the TGS to issue a service ticket for Service B.
- The TGS confirms the validity of Client A's ticket and the ability to issue service tickets for Service B.
- The TGS issues a service ticket containing a new session key for use by Client A and Service B

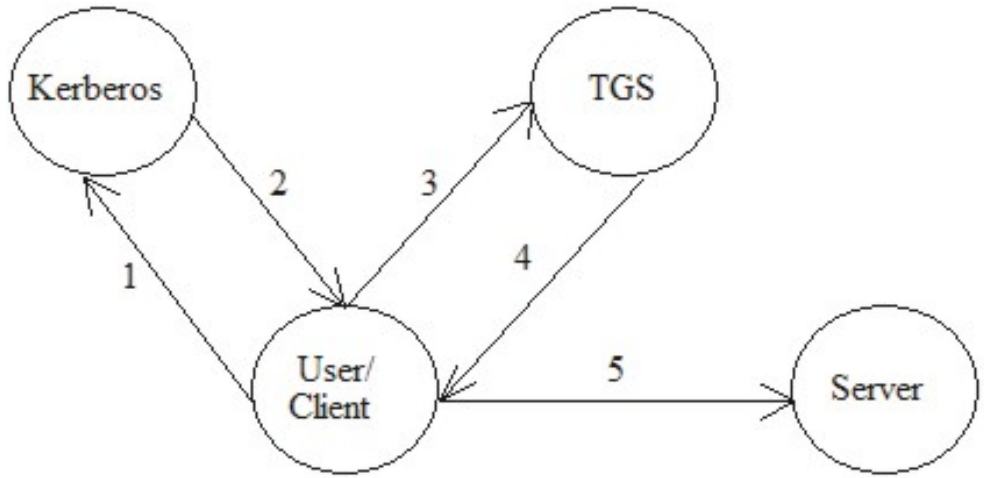
secureendpoints

The Application Service

- The Service Ticket includes an envelope encrypted with Service B's secret key stored in the Kerberos DB. This envelope includes a copy of the session key, Client A's name, and a window of validity
- Using the session key, Service B can perform a challenge-response exchange with Client A to mutually authenticate the two entities.
- Service B validates the rights of Client A against an authorization service (not Kerberos)

secureendpoints

Kerberos Protocol Exchange



1. Request for TGS ticket
2. Ticket for TGS
3. Request for Server ticket
4. Ticket for Server
5. Request for service

secureendpoints

Authentication vs Authorization

- Kerberos provides name based authentication
- The granting of a TGT or Service Ticket to a Client does not demonstrate any rights to access a service or perform an action
- Authorization Services specify which privileges a particular entity may perform

secureendpoints

What Authorization Mechanisms are used with Kerberos?

- Local host databases (/etc/passwd)
- User account databases (~/.k5login)
- LDAP
- Site specific
- Microsoft or DCE PAC (embedded within Kerberos Ticket)

secureendpoints

Other Properties of Tickets

- Tickets may be flagged with special properties:
  - Forwardable and Forwarded
  - Proxiable and Proxy (useless)
  - May Postdate and Postdated
  - Invalid
  - Renewable
  - Initial (for TGT and Change Password)
  - Hardware Authentication
  - Pre-authenticated
  - Transited Path Checked (used with cross-realm)

Common Services which use Kerberos for Authentication

- Logon Service (PAM module or `login`)
- Secure Shell
- FTP
- CVS
- LDAP
- CIFS
- LPR
- AFS
- NFS
- TLS
- HTTP
- SMTP
- IMAP

What is GSS-API?

- The Generic Security Service API provides a security service neutral API for use by applications in order to enable them to be unaware of the underlying security service used by the operating environment.
- GSS provides standard bindings in C and Java
- GSS mechanisms such as Kerberos 5 define mechanism specific wire protocols allowing for cross-platform interoperability
- Microsoft's Kerberos SSP implements the GSS Kerberos 5 wire protocol

secureendpoints

GSS Pros and Cons

- Pros
  - Cross platform
  - Hides details from application developer
  - All implementations provide the same API
- Cons
  - Mechanism specific features are inaccessible
  - No common API for selecting the credentials to use
  - GSS API C Bindings are not thread safe

secureendpoints

Using GSS is recommended

- There is no standard Kerberos 5 API
  - MIT, Heimdal, and others each have their own
- Solaris and Windows only export the GSS or SSP interface
- IETF will (soon) form a new working group to improve GSS
  - Look for the announcement of the Kitten working group (daughter of CAT – Common Authentication Technologies)

secureendpoints

An Overview of MIT's Kerberos products

- MIT Kerberos 5 for Unix/Linux
- Kerberos for Mac OS X (ships with OS)
- Kerberos for Windows

secureendpoints

MIT Kerberos 5 for Unix/Linux

- Kerberos 5 library
- Kerberos 4 compatibility library
- GSS-API Kerberos 5 library
- Kerberized Apps (Telnet, FTP, r-cmd, login)
- Key Distribution Center
- Kerberos Admin Service
- Kerberos 5 to 4 Service

secureendpoints

Kerberos for Mac OS X

- MIT Kerberos 5 is Apple's single sign-on solution
- Mac OS X Server integrates MIT KDC
- Many Kerberized applications
  - Login
  - Mail
  - Apple File Protocol
  - SSH
  - Screen saver
  - ftp (Server)
  - Active Directory plug-in

secureendpoints

Kerberos for Mac OS X (cont.)

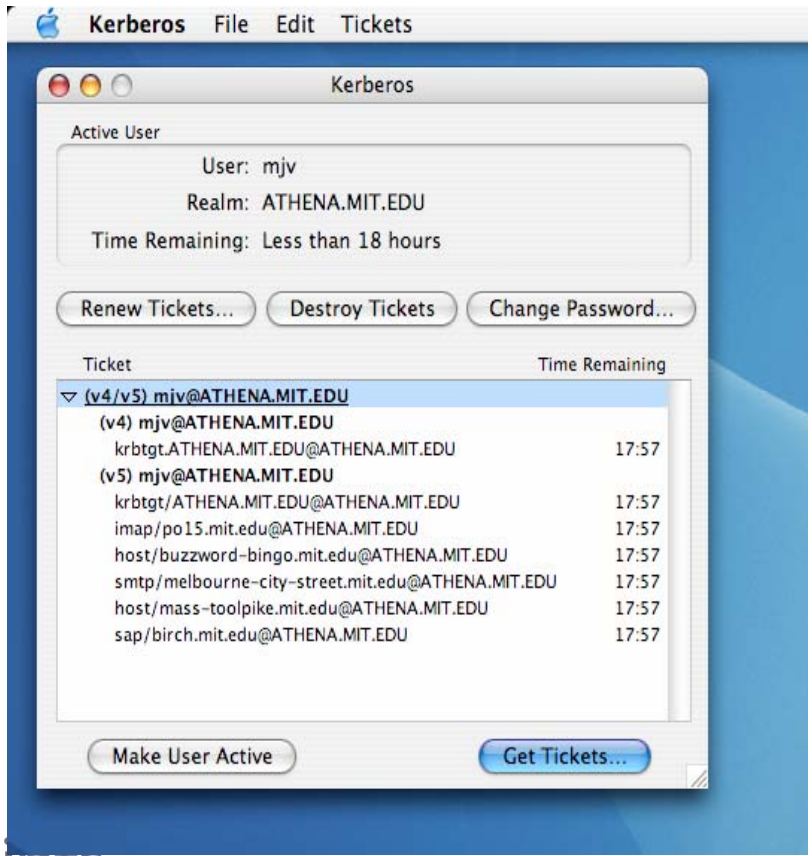
- A superset of the Unix/Linux offering:
  - CCAPI based memory cache; effortlessly supports multiple credentials
  - Stable API & ABI
  - Integration with Mac OS X's security model
  - Integrated UI tools take advantage of Mac OS X features
  - Kerberos Login Library API for initial ticket acquisition
  - Kerberos v5 and v4 tickets stored in the same cache
  - End users isolated from Kerberos v4 and v5 differences

secureendpoints

Mac OS X Obtain Ticket Dialog

secureendpoints

Mac OS X Klist



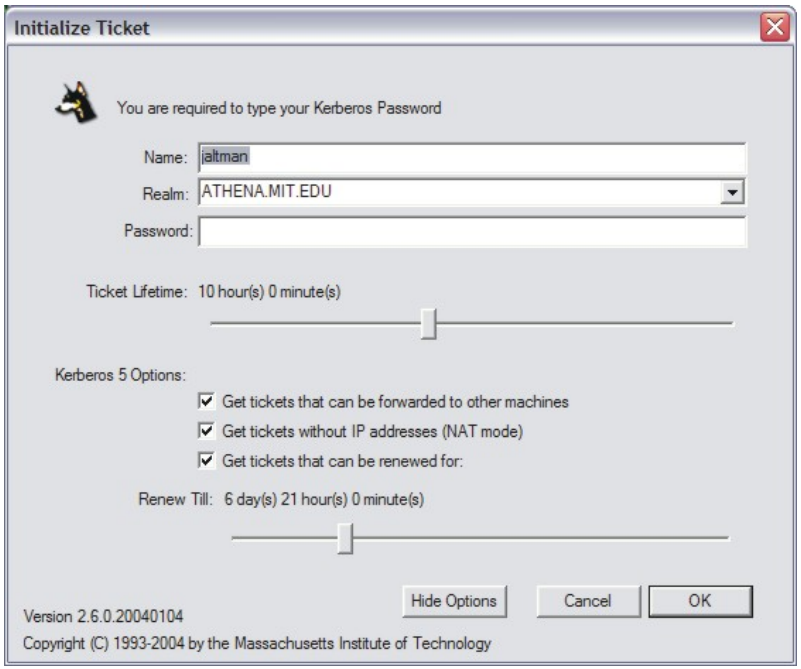
secureendpoints

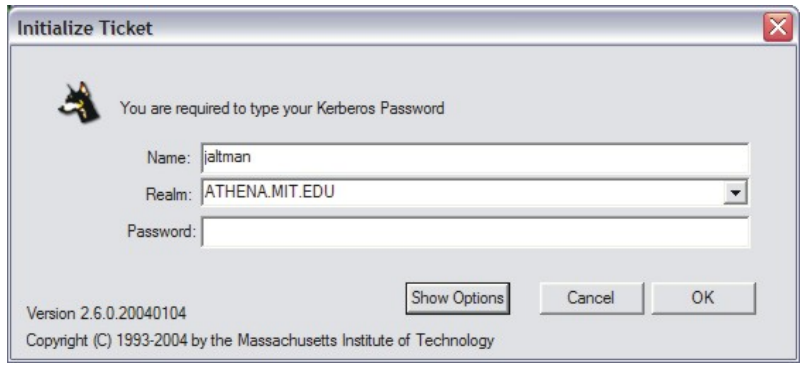
MIT Kerberos for Windows

- A superset of the Unix/Linux offering:
  - CCAPI based memory cache; effortlessly supports multiple credentials
  - Stable API & ABI
  - Integrated with Microsoft's Kerberos LSA Authentication Provider for a Single Sign-on solution.
  - Provides the Kerberos API Microsoft chose not to export to developers.
  - Leash Ticket Manager obtains and auto-renews Kerberos tickets as well as AFS tokens

secureendpoints

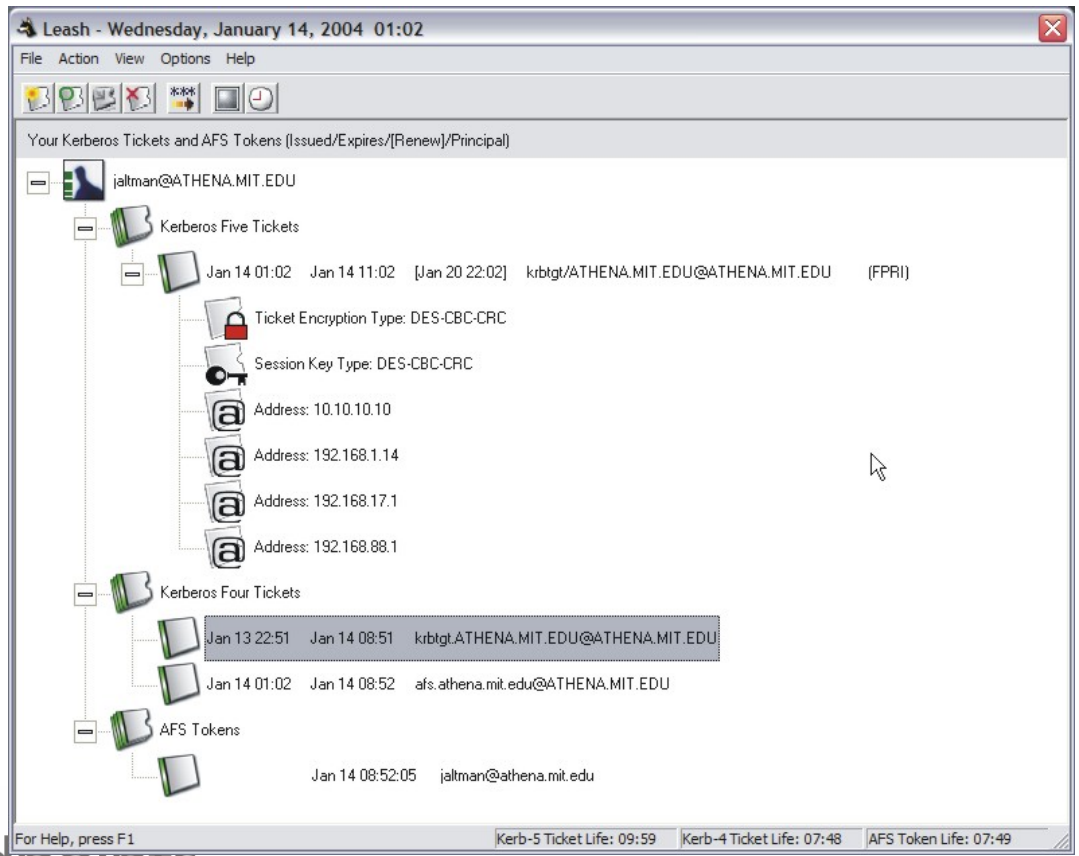
KFW Leash Ticket Manager Obtain Ticket Dialog





secureendpoints

KFW Leash Ticket Manager Ticket List



secureendpoints

MIT Kerberos Configuration File Basics

- Realm Configuration Data
  - KDC(s)
  - password change
  - krb524
  - Ticket Encryption Type restrictions (do not use)
- Realm to Domain/Hostname mappings
- Library defaults
  - Local Realm
  - Ticket Flags
  - Addressless Tickets
  - DNS lookup for Servers and Realms
- Cross-realm Authentication Paths

Realm Configuration

```
[realms]
        ATHENA.MIT.EDU = {
                kdc = kerberos.mit.edu
                kdc = kerberos-1.mit.edu
                kdc = kerberos-2.mit.edu
                kdc = kerberos-3.mit.edu
        }
        CC.COLUMBIA.EDU = {
                kdc = kerberos.cc.columbia.edu
        }
        DEMENTIA.ORG = {
                default_domain = dementia.org
                kdc = meredith.dementia.org
                kdc = alycia.dementia.org
                master_kdc = meredith.dementia.org
        }
        GRAND.CENTRAL.ORG = {
                default_domain = central.org
                kdc = penn.central.org
                kdc = grand-opening.mit.edu
        }
        OPENAFS.ORG = {
                admin_server = virtue.openafs.org
                default_domain = openafs.org
                kdc = virtue.openafs.org
                master_kdc = virtue.openafs.org
        }
```

```
RAEBURN.ORG = {
                default_domain = raeburn.org
        }
        SDE.CC = {
                admin_server = jude.sde.cc
                default_domain = sde.cc
                kdc = jude.sde.cc
                krb524_server = jude.sde.cc:4444
                krb524_server = jude.sde.cc:44445
                master_kdc = jude.sde.cc
        }
        SECURE-ENDPOINTS.COM = {
                admin_server = redhat71.secure-endpoints.com
                default_domain = secure-endpoints.com
                kdc = redhat71.secure-endpoints.com
                kdc = secure-endpoints.com
                master_kdc = redhat71.secure-endpoints.com
        }
        WINDOWS.SECURE-ENDPOINTS.COM = {
                admin_server = dc.windows.secure-endpoints.com
                default_domain = windows.secure-endpoints.com
                kdc = dc.windows.secure-endpoints.com
                master_kdc = dc.windows.secure-endpoints.com
        }
```

Domain to Realm Mappings

- [domain_realm]
- .central.org = GRAND.CENTRAL.ORG
- .columbia.edu = CC.COLUMBIA.EDU
- .kermit.columbia.edu = KRB5.COLUMBIA.EDU
- .mit.edu = ATHENA.MIT.EDU
- .raeburn.org = RAEBURN.ORG
- .sde.cc = SDE.CC
- .secure-endpoints.com = SECURE-ENDPOINTS.COM
- .windows.secure-endpoints.com = WINDOWS.SECURE-ENDPOINTS.COM
- grand-opening.mit.edu = GRAND.CENTRAL.ORG
- grand.central.org = ANDREW.CMU.EDU

Library Defaults

- [libdefaults]
- default_realm = ATHENA.MIT.EDU
- default_tgs_enctypes = arcfour-hmac-md5 aes256-cts-hmac-sha1-96 aes128-cts-hmac-sha1-96 des3-cbc-sha1 des-cbc-crc des-cbc-md5 des-cbc-md4
- default_tkt_enctypes = arcfour-hmac-md5 aes256-cts-hmac-sha1-96 aes128-cts-hmac-sha1-96 des3-cbc-sha1 des-cbc-crc des-cbc-md5 des-cbc-md4
- dns_lookup_kdc = true
- dns_lookup_realm = true
- forwardable = false
- noaddresses = true
- proxiable = false

Cross-realm authentication paths (part 1)

```
[capaths]
        RAEBURN.ORG = {
                ATHENA.MIT.EDU = .
                ANDREW.CMU.EDU = DEMENTIA.ORG
                DEMENTIA.ORG = ATHENA.MIT.EDU
                GRAND.CENTRAL.ORG = ATHENA.MIT.EDU
                GRAND.CENTRAL.ORG = DEMENTIA.ORG
                OPENAFS.ORG = ATHENA.MIT.EDU
                OPENAFS.ORG = DEMENTIA.ORG
                SECURE-ENDPOINTS.COM = ATHENA.MIT.EDU
                SECURE-ENDPOINTS.COM = DEMENTIA.ORG
                WINDOWS.SECURE-ENDPOINTS.COM = ATHENA.MIT.EDU
                WINDOWS.SECURE-ENDPOINTS.COM = DEMENTIA.ORG
                WINDOWS.SECURE-ENDPOINTS.COM = SECURE-ENDPOINTS.COM
        }
        ATHENA.MIT.EDU = {
                ANDREW.CMU.EDU = DEMENTIA.ORG
                DEMENTIA.ORG = .
                GRAND.CENTRAL.ORG = DEMENTIA.ORG
                OPENAFS.ORG = DEMENTIA.ORG
                SECURE-ENDPOINTS.COM = DEMENTIA.ORG
                WINDOWS.SECURE-ENDPOINTS.COM = ATHENA.MIT.EDU
                WINDOWS.SECURE-ENDPOINTS.COM = SECURE-ENDPOINTS.COM
        }
        SECURE-ENDPOINTS.COM = {
                ANDREW.CMU.EDU = DEMENTIA.ORG
                ATHENA.MIT.EDU = DEMENTIA.ORG
                DEMENTIA.ORG = .
                RAEBURN.ORG = DEMENTIA.ORG
                RAEBURN.ORG = ATHENA.MIT.EDU
                GRAND.CENTRAL.ORG = DEMENTIA.ORG
                OPENAFS.ORG = DEMENTIA.ORG
                WINDOWS.SECURE-ENDPOINTS.COM = .
        }
```

Cross-realm authentication paths (part 2)

```
        WINDOWS.SECURE-ENDPOINTS.COM = {
                RAEBURN.ORG = ATHENA.MIT.EDU
                RAEBURN.ORG = SECURE-ENDPOINTS.COM
                RAEBURN.ORG = DEMENTIA.ORG
                ANDREW.CMU.EDU = SECURE-ENDPOINTS.COM
                ANDREW.CMU.EDU = DEMENTIA.ORG
                ATHENA.MIT.EDU = SECURE-ENDPOINTS.COM
                ATHENA.MIT.EDU = DEMENTIA.ORG
                DEMENTIA.ORG = SECURE-ENDPOINTS.COM
                GRAND.CENTRAL.ORG = SECURE-ENDPOINTS.COM
                GRAND.CENTRAL.ORG = DEMENTIA.ORG
                OPENAFS.ORG = SECURE-ENDPOINTS.COM
                OPENAFS.ORG = DEMENTIA.ORG
                SECURE-ENDPOINTS.COM = .
        }
GRAND.CENTRAL.ORG = {
                ATHENA.MIT.EDU = .
                DEMENTIA.ORG = ATHENA.MIT.EDU
                SECURE-ENDPOINTS.COM = ATHENA.MIT.EDU
                SECURE-ENDPOINTS.COM = DEMENTIA.ORG
                OPENAFS.ORG = ATHENA.MIT.EDU
                OPENAFS.ORG = DEMENTIA.ORG
                WINDOWS.SECURE-ENDPOINTS.COM = ATHENA.MIT.EDU
                WINDOWS.SECURE-ENDPOINTS.COM = DEMENTIA.ORG
                WINDOWS.SECURE-ENDPOINTS.COM = SECURE-ENDPOINTS.COM
}
```

Kadmin Entries for Principals

- K/M@SECURE-ENDPOINTS.COM
  - Master key for the entire Kerberos Database
- afs@SECURE-ENDPOINTS.COM
  - AFS service principal for secure-endpoints.com cell
- host/redhat71.secure-endpoints.com@SECURE-ENDPOINTS.COM
  host/vmware-xp.secure-endpoints.com@SECURE-ENDPOINTS.COM
  - Host keys used to authenticate user logins both local and remote
- jaltman@SECURE-ENDPOINTS.COM
  - My user principal in SECURE-ENDPOINTS.COM
- krbtgt/DEMENTIA.ORG@SECURE-ENDPOINTS.COM
  krbtgt/SECURE-ENDPOINTS.COM@DEMENTIA.ORG
  - Cross realm trust keys for use with Heimdal realm
- krbtgt/SECURE-ENDPOINTS.COM@WINDOWS.SECURE-ENDPOINTS.COM
  krbtgt/WINDOWS.SECURE-ENDPOINTS.COM@SECURE-ENDPOINTS.COM
  - Cross realm trust keys for use with Windows domain

secureendpoints

User Principal Entry

- Principal: jaltman@SECURE-ENDPOINTS.COM
- Expiration date: [never]
- Last password change: Tue Feb 17 13:39:30 EST 2004
- Password expiration date: [none]
- Maximum ticket life: 7 days 00:00:00
- Maximum renewable life: 120 days 00:00:00
- Last modified: Tue Feb 17 13:39:30 EST 2004 (jaltman/admin@SECURE-ENDPOINTS.COM)
- Last successful authentication: [never]
- Last failed authentication: [never]
- Failed password attempts: 0
- Number of keys: 6
- Key: vno 1, Triple DES cbc mode with HMAC/sha1, no salt
- Key: vno 1, AES-256 CTS mode with 96-bit SHA-1 HMAC, no salt
- Key: vno 1, AES-128 CTS mode with 96-bit SHA-1 HMAC, no salt
- Key: vno 1, ArcFour with HMAC/md5, no salt
- Key: vno 1, DES cbc mode with CRC-32, no salt
- Key: vno 1, DES cbc mode with CRC-32, Version 4
- Attributes: REQUIRES_PRE_AUTH
- Policy: [none]

Heimdal Cross-realm Principal

- Principal: krbtgt/DEMENTIA.ORG@SECURE-ENDPOINTS.COM
- Expiration date: [never]
- Last password change: Thu Apr 15 18:31:09 EDT 2004
- Password expiration date: [none]
- Maximum ticket life: 7 days 00:00:00
- Maximum renewable life: 120 days 00:00:00
- Last modified: Thu Apr 15 18:31:09 EDT 2004 (root/admin@SECURE-ENDPOINTS.COM)
- Last successful authentication: [never]
- Last failed authentication: [never]
- Failed password attempts: 0
- Number of keys: 2
- Key: vno 1, DES cbc mode with CRC-32, Version 4
- Key: vno 1, Triple DES cbc mode with HMAC/sha1, no salt
- Attributes: REQUIRES_PRE_AUTH
- Policy: [none]

secureendpoints

Windows Cross-realm Principal

- Principal: krbtgt/WINDOWS.SECURE-ENDPOINTS.COM@SECURE-ENDPOINTS.COM
- Expiration date: [never]
- Last password change: Tue Feb 17 17:02:04 EST 2004
- Password expiration date: [none]
- Maximum ticket life: 7 days 00:00:00
- Maximum renewable life: 120 days 00:00:00
- Last modified: Tue Feb 17 17:02:04 EST 2004 (jaltman/admin@SECURE-ENDPOINTS.COM)
- Last successful authentication: [never]
- Last failed authentication: [never]
- Failed password attempts: 0
- Number of keys: 2
- Key: vno 2, ArcFour with HMAC/md5, no salt
- Key: vno 2, DES cbc mode with CRC-32, no salt
- Attributes: REQUIRES_PRE_AUTH
- Policy: [none]

AFS Service Principal

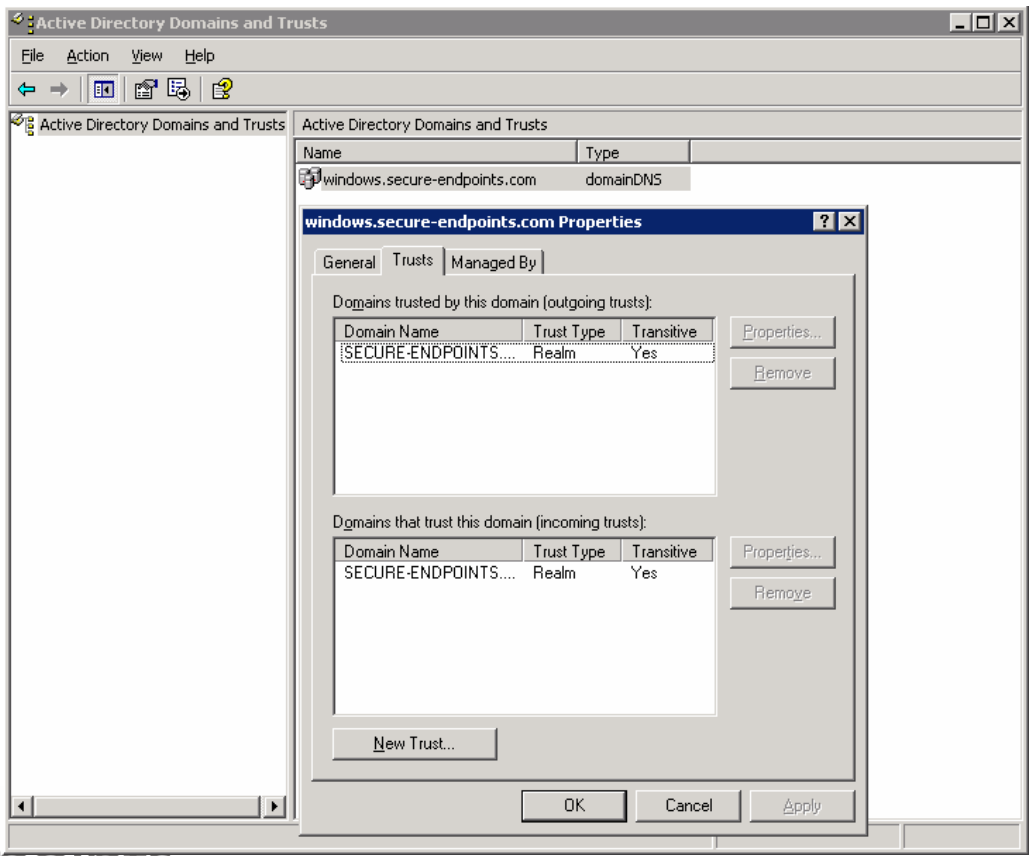
- Principal: afs@SECURE-ENDPOINTS.COM
- Expiration date: [never]
- Last password change: Thu Apr 01 18:09:52 EST 2004
- Password expiration date: [none]
- Maximum ticket life: 7 days 00:00:00
- Maximum renewable life: 120 days 00:00:00
- Last modified: Thu Apr 01 18:30:48 EST 2004 (root/admin@SECURE-ENDPOINTS.COM)
- Last successful authentication: [never]
- Last failed authentication: [never]
- Failed password attempts: 0
- Number of keys: 1
- Key: vno 1, DES cbc mode with CRC-32, no salt
- Attributes:
- Policy: [none]

Host Service Principal for use in Cross-realm environment with Windows Domain

- Principal: host/redhat71.secure-endpoints.com@SECURE-ENDPOINTS.COM
- Expiration date: [never]
- Last password change: Fri Feb 20 22:54:13 EST 2004
- Password expiration date: [none]
- Maximum ticket life: 7 days 00:00:00
- Maximum renewable life: 120 days 00:00:00
- Last modified: Fri Feb 20 22:54:13 EST 2004 (jaltman/admin@SECURE-ENDPOINTS.COM)
- Last successful authentication: [never]
- Last failed authentication: [never]
- Failed password attempts: 0
- Number of keys: 2
- Key: vno 7, ArcFour with HMAC/md5, no salt
- Key: vno 7, DES cbc mode with CRC-32, no salt
- Attributes:
- Policy: [none]

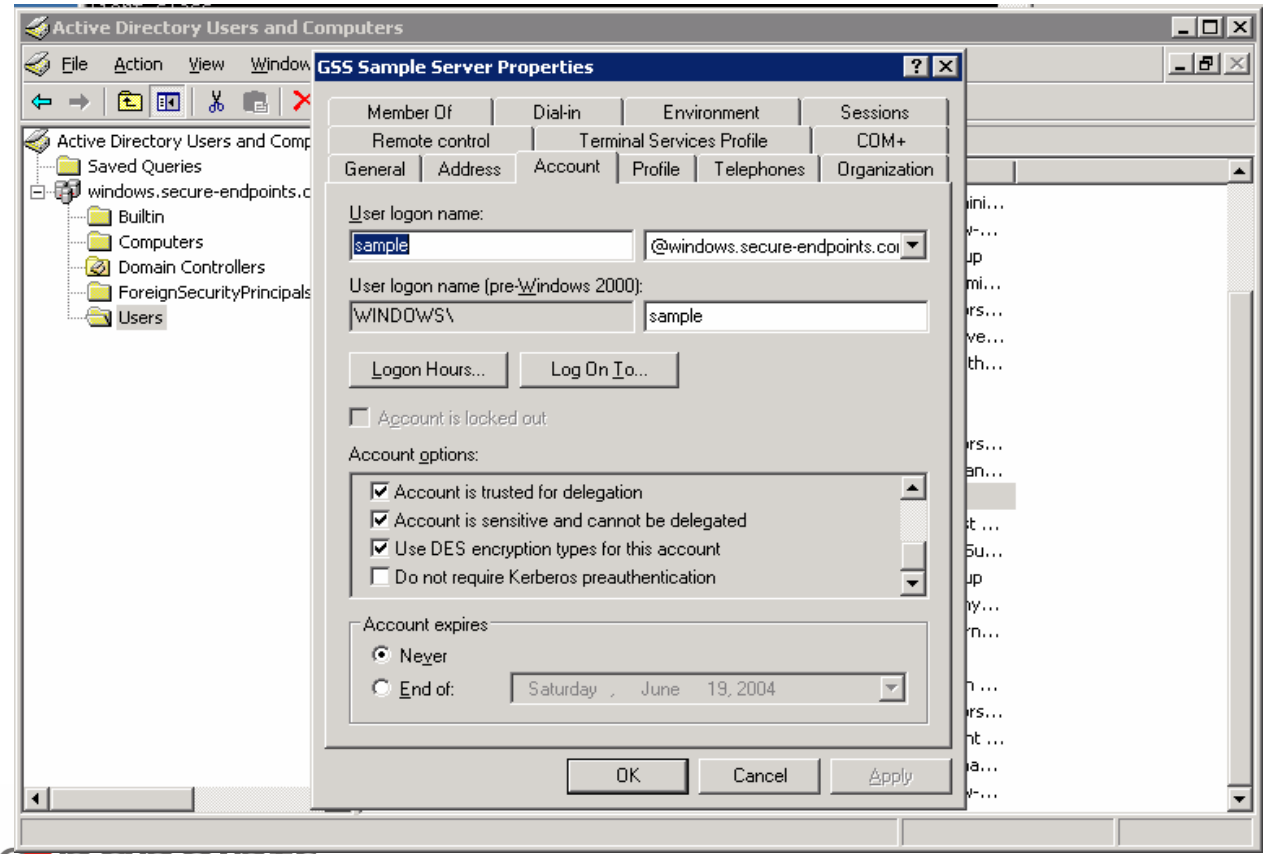
secureendpoints

Setting Up Cross-realm Trusts with Windows



secureendpoints

Configure Service Principals with Windows



secureendpoints

Interoperability Issues: Encryption Types

- MIT: DES, 3DES, RC4, AES
- Windows: RC4 for Windows; DES-CBC-MD5 for non-Windows (future versions will support RC4 for non-Windows)
- Java: DES

secureendpoints

Interoperability Issues: Name Canonicalization

- Kerberos is case sensitive
- Windows is case insensitive
- Window has multiple service principal names for the same service depending on context

secureendpoints

Interoperability Issues: Credential Cache sharing

- Single Sign-on requires credentials be obtained at logon and shared by all applications
- There is no single credential cache API supported by MIT, Windows, and Java
  - MIT KFW's "MSLSA:" credential cache is a pass-through to the Windows LSA cache
  - Java provides a JAAS Login Provider which can obtain credentials using Windows LSA using username/password
  - Windows and Java cannot access MIT "CCAPI:" caches which are used when multiple simultaneous principals are required
- If your only TGT is RC4 and you attempt to execute a Java program which requires Kerberos service tickets, it will lose.

secureendpoints

Environmental Issues

- Network Address Translation
  - Tickets bound to addresses do not work
  - Channel Bindings using addresses do not work
- Firewalls
  - Kerberos ports are often mistakenly blocked by ISPs
    - Krb524 port 4444
      a windows worm used this port for tcp although krb524 only uses udp it is blocked anyway
    - KDC port 88
      ISPs blocking all Windows ports block 88 because it is used by Windows

secureendpoints

Demos and Code Walk-through

- Various GSS Client and Server combinations
  - MIT Kerberos
  - Microsoft Windows SSP
  - Java GSS

secureendpoints

GSS Sample Client to SSP Sample Server Configuration

- Create service account in the Windows Server Active Directory
  - Use the Active Directory Users and Computers Administrative Tool to create a new user account.
  - Assign a user logon name to the account without spaces
  - Assign a password to the account
  - Select the following Account options:
    - "Use DES encryption types for the account"
    - "Account is trusted for delegation"
    - "Account is sensitive and cannot be delegated“
    - "User cannot change password“
    - "Password never expires
- Assign a Service Principal Name (SPN) to the account utilizing the SETSPN.EXE program installed from the Operating System's Support Tools folder on the original CD. An SPN is usually of the form service-name/hostname but may be of the form service-name/hostname/domain depending on the deployment. There can be multiple SPNs assigned to a single account. For this example specify a <service-name> and a <hostname> where the <service-name> is arbitrary and the <hostname> is the fully qualified domain name of the machine on which the SSP sample server will be executed.
  - SETSPN -A <SPN> <account-name>

Testing GSS Sample Client to SSP Sample Server

- Start gssserver on Windows specifying the selected options of your choice:
  - gssserver.exe [options] <service-name> <password> <REALM>
- On the machine to be used to execute the MIT gss-client:
  - Using "kinit" obtain a Ticket Getting Ticket (TGT) for a client principal capable of obtaining a service ticket for the service principal <service-name>/<hostname>@<REALM>
  - Execute the MIT gss-client specifying the selected options of your choice:
    - gss-client [options] <hostname> <service-name> <test-message>
- After successful delivery of the test-message, executing "klist" on the client machine will display:
  - the client principal's TGT "krbtgt/REALM@REALM“
  - the service ticket for the gssserver service principal
  - "service-name/hostname@REALM"

secureendpoints

IETF Kerberos Working Group

- http://www.ietf.org/html.charters/krbwg-charter.html
- Recent Documents
  - Kerberos 5 Clarifications
  - Kerberos 5 Crypto Framework
  - AES for Kerberos 5
  - AES for GSSAPI Kerberos 5 Mechanism
- In Progress Documents
  - Public Key Initialization for Kerberos 5
  - Kerberos 5 Extensions (will replace Kerberos 5)
    - Internationalization
    - Extensibility
  - Generalized Framework for Pre-authentication

secureendpoints

MIT Kerberos Futures

- 1.4 Release Targeted for Summer
- New KFM in Mac OS X (Tiger)
- New KFW after 1.4 release

secureendpoints

Your Donations Can Help MIT Kerberos Succeed

- MIT Kerberos is the leading voice in Kerberos standards development
  - provides reference implementation
  - works with many vendors to help ensure interoperability of Kerberos related technologies
  - distributes cross-platform tools for developers
  - Supports end users and developers via mailing lists and newsgroups
- Costs associated with this level of commitment have significantly increased as Kerberos has become a must have technology
- Donations to MIT Kerberos are tax deductible

secureendpoints

Q & A

You ask, I answer

Important URLs

- MIT Kerberos
  http://web.mit.edu/kerberos
- Microsoft Kerberos
  http://www.microsoft.com/kerberos/
- Good Java GSS links
  http://www-106.ibm.com/developerworks/java/library/j-gss-sso
  http://java.sun.com/j2se/1.4.2/docs/guide/security/jgss/single-signon.html

secureendpoints

secure endpoints

Contact Information

Jeffrey Eric Altman
jaltman * secure-endpoints * com
http://www.secure-endpoints.com