nexus enlightenment etu

Hadoop Security Overview
- From security infrastructure deployment to high-level services

Jason Shih
Etu

30 Nov, 2012
Hadoop & BigData Technology Conference

Outline

- **Kerberos & LDAP**
  - Configuration & Installation
  - Authentication & Authorization
  - Interoperability
- **Hadoop Security & Services**
  - Authentication & Authorization in Hadoop
  - Token Delegation & communication path
  - *HDFS: **NN & DN***
  - *MapReduce: **JT+TT***
  - *HBase: **ZK+MASTER+RS***
- **Etu Appliance**
  - **New features & key benefits**
  - **Software stacks, versions & HW spec.**
- **Troubleshooting**

etu
nexus enlightenment

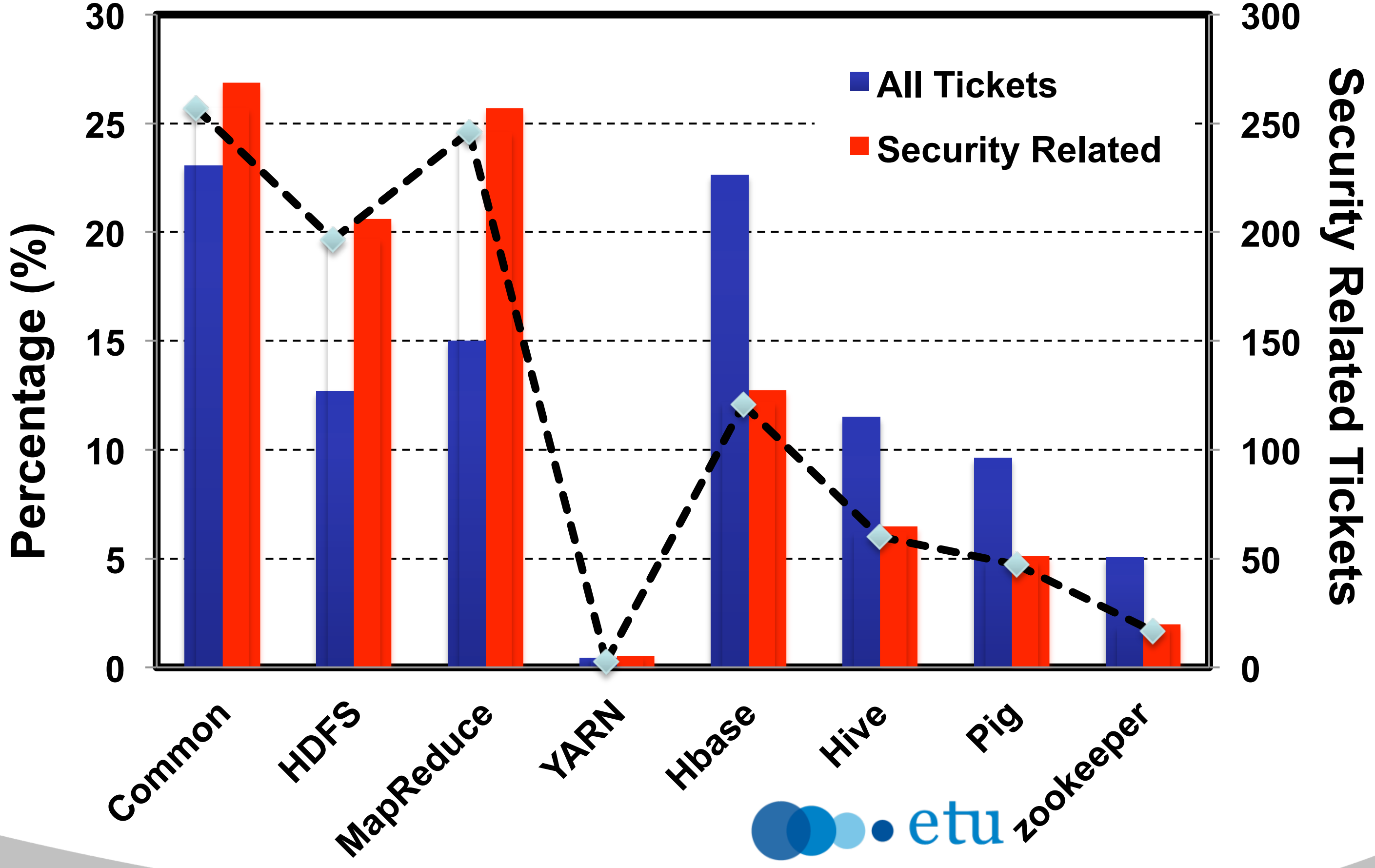
Who am I?

- Etu
  - Hadoop System Architect
- Grid Computing Centre, ASGC
  - Tech Lead on Grid Operation
  - Scope: DC, OP, DM & GT
  - Experiment Support (LHC Analysis Software, ES, EC (W&C) etc.)
- Before Grid Computing – HPC @ ASCC
  - System administration (IBM, SGI, Sun, *nix)
  - Architecture Design & Parallel filesystem
  - Performance Tuning & Optimization
  - Application Support etc.

etu
nexus enlightenment

Does security matter?

- **Ticket Breakdown:**
  - Comprise ~3.1% issues are security related
    - Hadoop common, HDFS, MR, YARN, HBase, Hive, & Pig etc.
  - Majority are common+HDFS+MR related: ~73%



etu
nexus enlightenment

HBTC 2012

LDAP

(lightweight) directory access protocol

Small bit of data, mostly read access

NIS

Pros: *setup, administration, widely support & scale fairly well*

Cons: *weakly encrypted password, difficult to FW, lack of system auth*

NIS+

Complicated, limited client support.

Kerberos & LDAP

Configuration & Installation

Authentication & Authorization

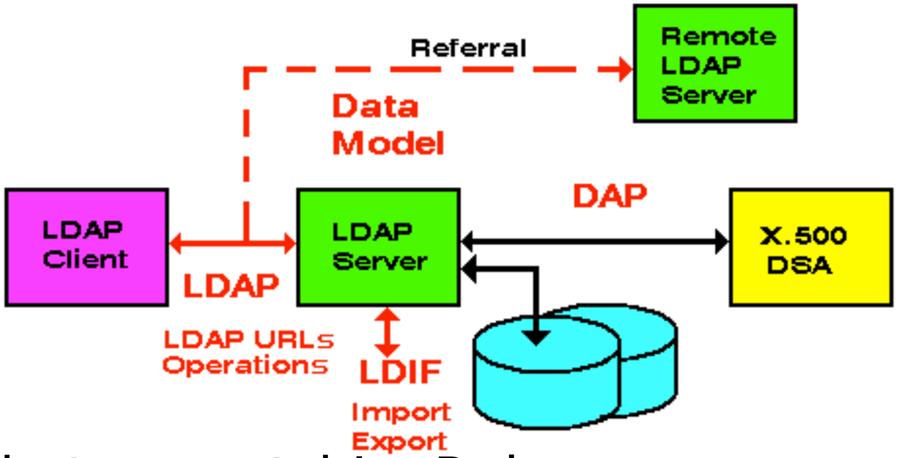
Interoperability

LDAP Authentication

- OpenLDAP: Lightweight Directory Access Protocol
  - X.500 base (model for directory service in OSI concept)
  - X.400 Std. by ITU late 70's & early 80's (email service)
- Why directory?
  - Specialized database design for frequent queries but infrequent updates
  - lack of rollback functionality & transaction support
  - Easily replicated aiming for high availability & scalability (but depend on size of info being published or replicated).

Terminology: http://www.zytrax.com/books/ldap/apd/

LDAP Overview



- Building blocks:
  - Schemas, objectClasses, Attributes, matchingRules, Operational objects etc.
- Models:
  - Information
    - information or data presented may/may-not the way data is actually stored
  - Naming:
    - def: 'dc=example,dc=com’ stumble across in LDAP
  - Functional
    - Read, Search, Write & Modify
  - Security
    - Fine grained manner, who can do what to what data

etu
nexus enlightenment

Kerberos Introduction

- What is Kerberos
  - Named after Cerberus, the three-headed dog of Greek mythology, because?
  - Composite by three components:
    - KDC (Kerberos Distribution Center)
    - Clients (Users/Hosts/Services)
    - Server (Service providers requested to establish session)
  - Scope of deployment: realm
  - KDC provide:
    - AS (Authentication Server)
    - TGS (Ticket Granting Service)

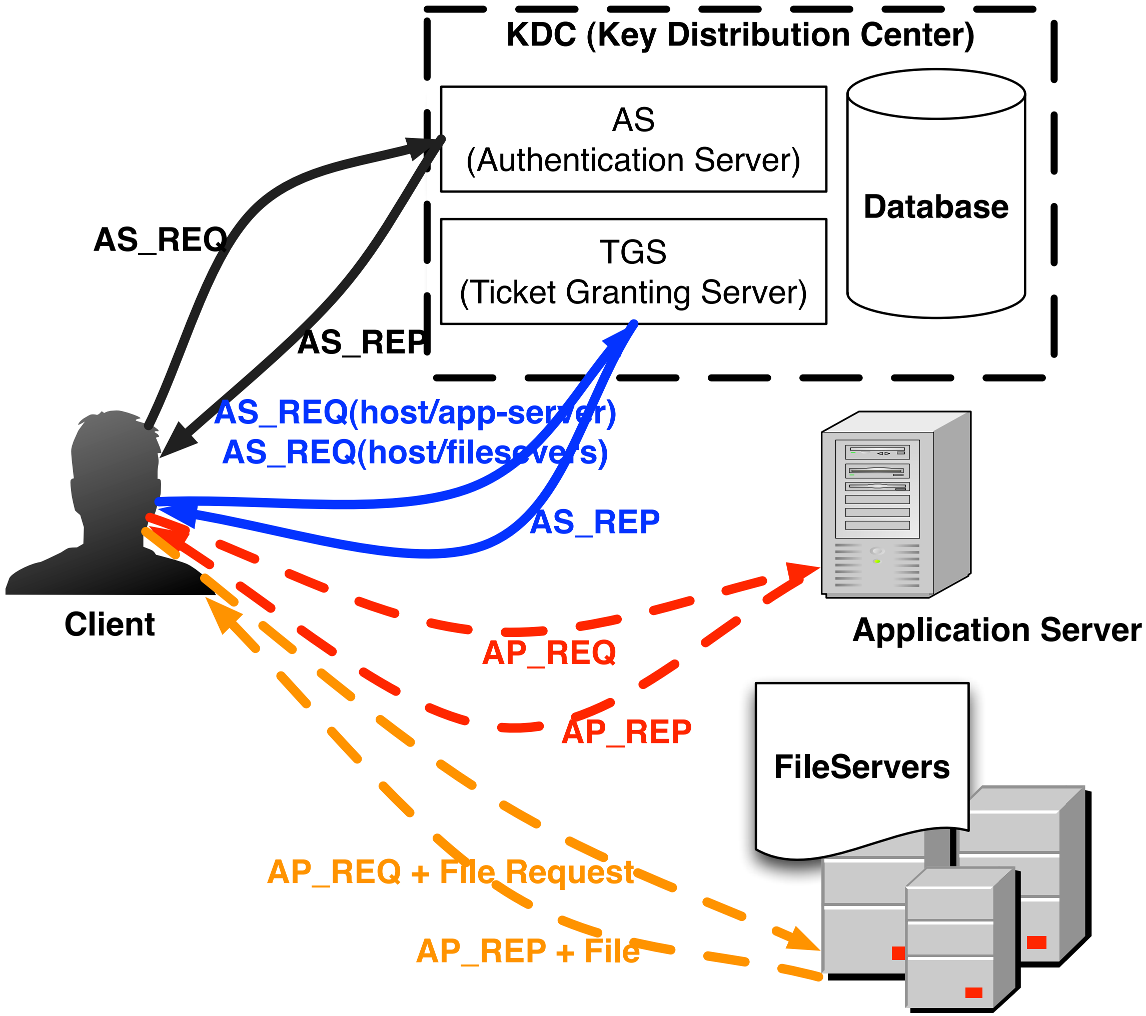
Kerberos Introduction (*cont'*)

- Kerberos Client
  - PAM enable (pam_krb5)
    - Other application, recompilation effort required: e.g. OpenSSH
  - Application w/ native Kerberos support but few limited to ver. IV
- Other Extension
  - Windows Authentication (AD)
  - NFS Authentication & Encryption
  - AFS (Global Filesystem)
- Symmetric key operations
  - Order of magnitude Faster than public key operations e.g. SSL
- Performs authentication not authorization
- When user authenticates, they are given a "ticket"
  - Default Lifetime: 8Hr

Kerberos: Definition & Terminology

- **KDC (Kerberos Distribution Center)**
- **TGT (Ticket Granting Ticket)**
  - Special ticket permit client to obtain additional Kerberos ticket within same realm
- **Keytab**
  - key table file containing one or more keys, same as for hosts & users
- **Principal**
  - Primary
    - First part of a Kerberos principal
    - User: username, Service: the name of the service
  - Instance
    - Provide information that qualifies the primary
    - User: desc. the intended use of corresponding credentials
    - Host: FQDN
  - Realm
    - Logical network served by a single Kerberos DB and a set of KDC

Kerberos Overview



Kerberos Principals & Realms

- Principal
  - Generic: Name/instance@realm
  - Examples
    - etu@testdomain.com
    - etu/admin
    - host/master.testdomain.com
    - ldap/ldap.testdomain.com
  - Realm
    - Typically domain name in all **CAPS:**
      e.g.: TESTDOMAIN.COM

etu
nexus enlightenment

Kerberos Command line

- Administration
  - – kadmin: used to make changes to the accounts in the Kerberos database
    - *kadmin.local*
  - – klist: used to view the tickets in the credential cache
  - – kinit: used to log onto the realm with the client’s key
  - – kdestroy: erases the credential cache
  - – kpasswd: used to change user passwords
  - – kprop: used to synch the master KDC with replicas, if any
- Utility
  - – kdb5_util: *create, destroy, stash, dump, load, ark, add_mkey, use_mkey, list_mkeys, update_princ_encryption & purge_mkeys*

Kerberos Administration (kadmin.local)

- Available requests:

  add_principal, addprinc, ank
  delete_principal, delprinc
  modify_principal, modprinc
  change_password, cpw
  get_principal, getprinc
  list_principals, listprincs, get_principals, getprincs
  add_policy, addpol
  modify_policy, modpol
  delete_policy, delpol
  get_policy, getpol
  list_policies, listpols, get_policies, getpols
  get_privs, getprivs
  ktadd, xst
  ktremove, ktrem
  lock
  unlock
  purgekeys

Kerberos Principals (I)

- **Default principals (default realm: TESTDOMAIN.COM)**

K/M@TESTDOMAIN.COM
hdfs@TESTDOMAIN.COM
kadmin/admin@TESTDOMAIN.COM
kadmin/changepw@TESTDOMAIN.COM
Kadmin/master.testdomain.com@TESTDOMAIN.COM
krbtgt/TESTDOMAIN.COM@TESTDOMAIN.COM
ldapadm@TESTDOMAIN.COM
ldap/master.testdomain.com@TESTDOMAIN.COM

Kerberos Principals (II)

- **Principals details *(KV no., expiration & attributes)***

```
Principal: hdfs@TESTDOMAIN.COM
Expiration date: [never]
Last password change: Thu Nov 15 19:44:31 CST 2012
Password expiration date: [none]
Maximum ticket life: 1 day 00:00:00
Maximum renewable life: 90 days 00:00:00
Last modified: Thu Nov 15 19:44:31 CST 2012 (kadmin/admin@TESTDOMAIN.COM)
Last successful authentication: [never]
Last failed authentication: [never]
Failed password attempts: 0
Number of keys: 5
Key: vno 2, aes128-cts-hmac-sha1-96, no salt
Key: vno 2, aes256-cts-hmac-sha1-96, no salt
Key: vno 2, arcfour-hmac, no salt
Key: vno 2, des3-cbc-sha1, no salt
Key: vno 2, des-cbc-crc, no salt
MKey: vno 1
Attributes:
Policy: [none]
```

Kerberos Server Configuration (I)

- **libdefaults**:

```
default_realm = TESTDOMAIN.COM
        ticket_lifetime = 48h
        renew_lifetime = 8760h
        forwardable = true
        proxiable = true
        default_tkt_enctypes = aes128-cts-hmac-sha1-96 …
        default_tgs_enctypes = aes128-cts-hmac-sha1-96 …
        permitted_enctypes = aes128-cts-hmac-sha1-96 …
        dns_lookup_realm = false
        dns_lookup_kdc = false
        allow_weak_crypto = 1
```

Allow_weak_crypto – temporary workaround

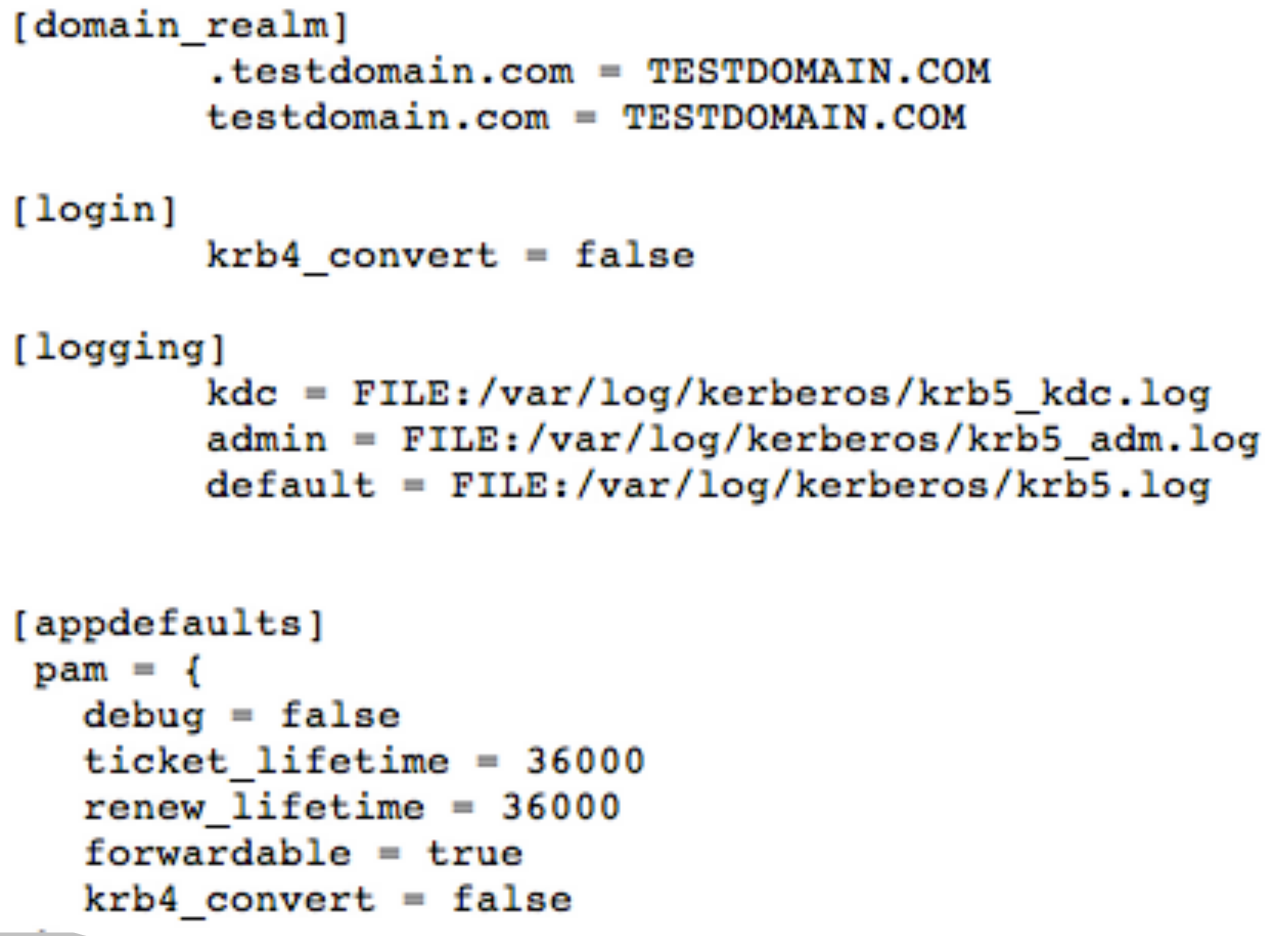
- **By default, clients & servers will not using keys for ciphers.**
- **Clients wont be able to authenticate to services w/ keys following support enctypes**
- **Zero downtime w/ service updating new/strong-cophers keys to keytab**
- **TGT can then update services' keys to a sets including keys w/ stronger ciphers (kadmin cpw -keepold command)**

Kerberos Server Configuration (II)

- **Realm & domain realm:**

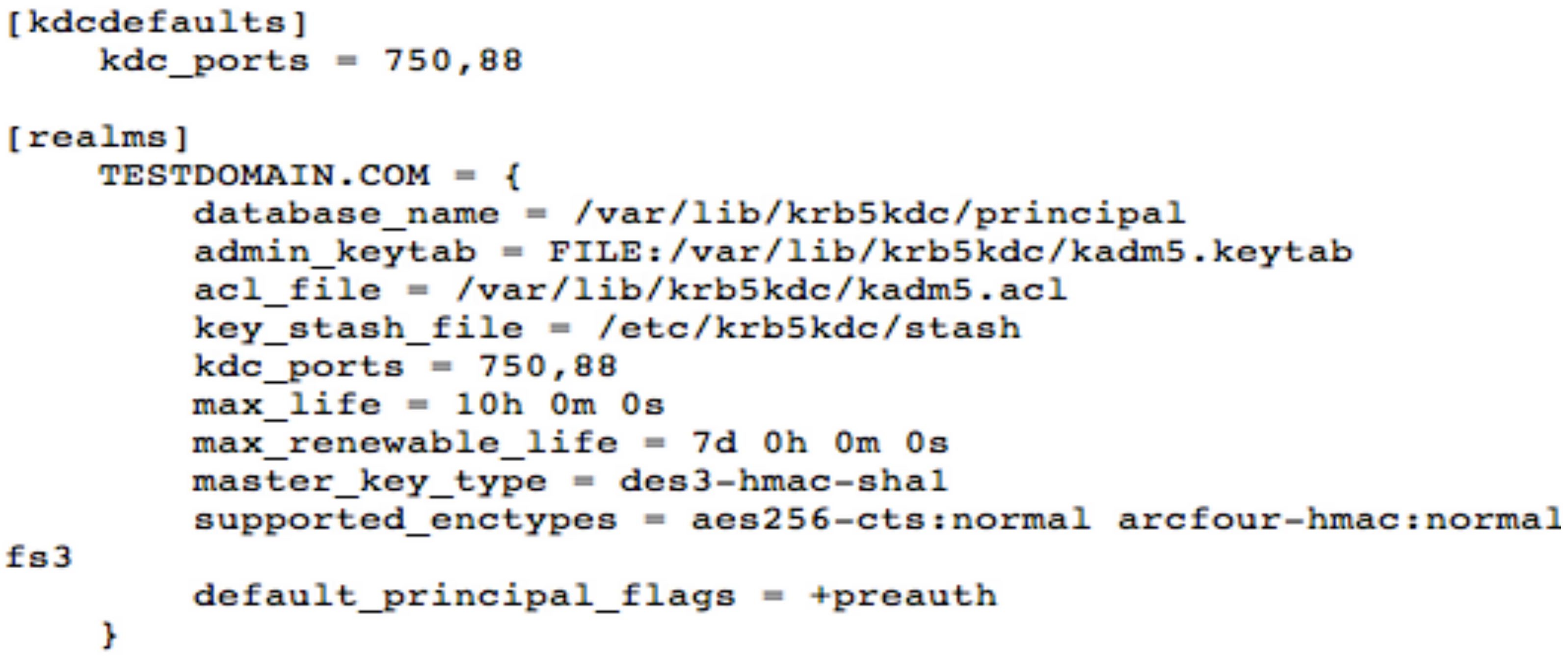
```
[realms]
        TESTDOMAIN.COM = {
                default_domain = testdomain.com
                kdc = etu-master.testdomain.com
                admin_server = etu-master.testdomain.com
                database_module = openldap_ldapconf
        }

[domain_realm]
        .testdomain.com = TESTDOMAIN.COM
        testdomain.com = TESTDOMAIN.COM
```

Kerberos Server Configuration (III)



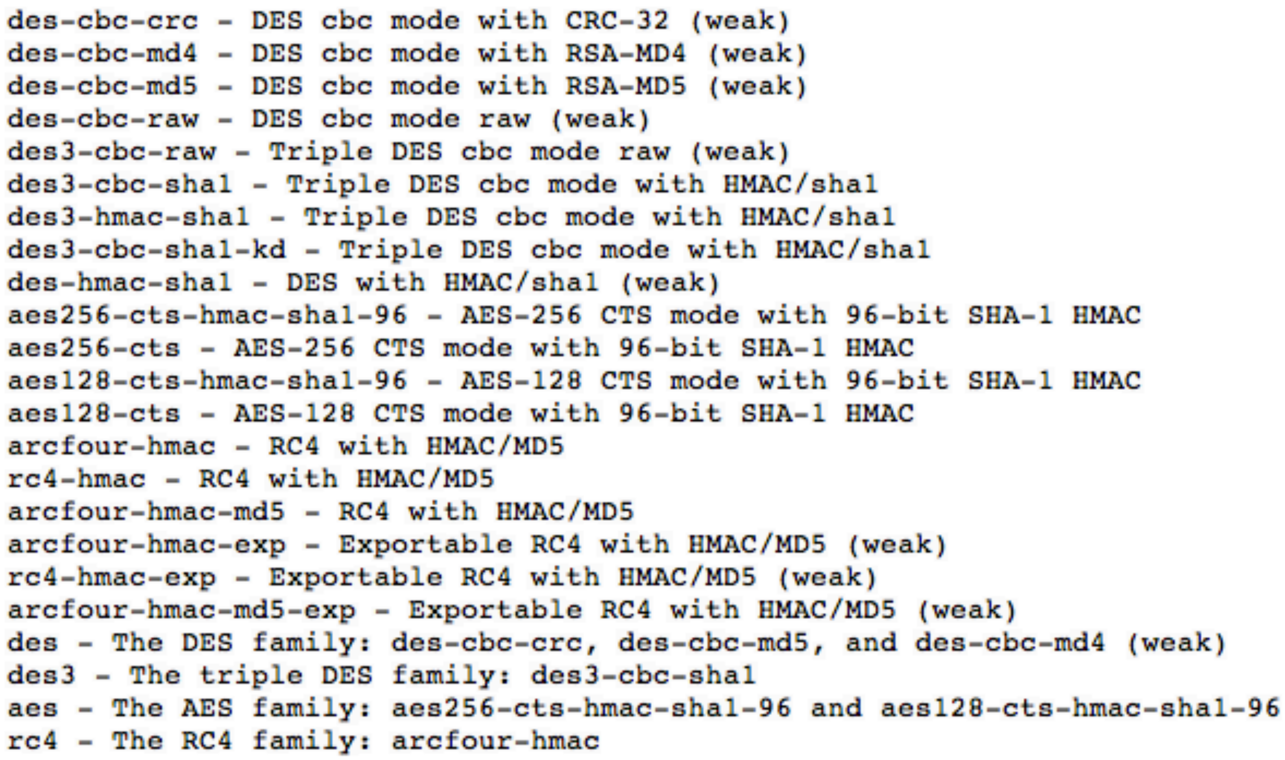
```
[domain_realm]
        .testdomain.com = TESTDOMAIN.COM
        testdomain.com = TESTDOMAIN.COM

[login]
        krb4_convert = false

[logging]
        kdc = FILE:/var/log/kerberos/krb5_kdc.log
        admin = FILE:/var/log/kerberos/krb5_adm.log
        default = FILE:/var/log/kerberos/krb5.log


[appdefaults]
 pam = {
   debug = false
   ticket_lifetime = 36000
   renew_lifetime = 36000
   forwardable = true
   krb4_convert = false
```

Kerberos KDC Config



```
[kdcdefaults]
    kdc_ports = 750,88

[realms]
    TESTDOMAIN.COM = {
        database_name = /var/lib/krb5kdc/principal
        admin_keytab = FILE:/var/lib/krb5kdc/kadm5.keytab
        acl_file = /var/lib/krb5kdc/kadm5.acl
        key_stash_file = /etc/krb5kdc/stash
        kdc_ports = 750,88
        max_life = 10h 0m 0s
        max_renewable_life = 7d 0h 0m 0s
        master_key_type = des3-hmac-sha1
        supported_enctypes = aes256-cts:normal arcfour-hmac:normal
fs3
        default_principal_flags = +preauth
    }
```

Kerberos Encryption Types

- 

```
des-cbc-crc - DES cbc mode with CRC-32 (weak)
des-cbc-md4 - DES cbc mode with RSA-MD4 (weak)
des-cbc-md5 - DES cbc mode with RSA-MD5 (weak)
des-cbc-raw - DES cbc mode raw (weak)
des3-cbc-raw - Triple DES cbc mode raw (weak)
des3-cbc-sha1 - Triple DES cbc mode with HMAC/sha1
des3-hmac-sha1 - Triple DES cbc mode with HMAC/sha1
des3-cbc-sha1-kd - Triple DES cbc mode with HMAC/sha1
des-hmac-sha1 - DES with HMAC/sha1 (weak)
aes256-cts-hmac-sha1-96 - AES-256 CTS mode with 96-bit SHA-1 HMAC
aes256-cts - AES-256 CTS mode with 96-bit SHA-1 HMAC
aes128-cts-hmac-sha1-96 - AES-128 CTS mode with 96-bit SHA-1 HMAC
aes128-cts - AES-128 CTS mode with 96-bit SHA-1 HMAC
arcfour-hmac - RC4 with HMAC/MD5
rc4-hmac - RC4 with HMAC/MD5
arcfour-hmac-md5 - RC4 with HMAC/MD5
arcfour-hmac-exp - Exportable RC4 with HMAC/MD5 (weak)
rc4-hmac-exp - Exportable RC4 with HMAC/MD5 (weak)
arcfour-hmac-md5-exp - Exportable RC4 with HMAC/MD5 (weak)
des - The DES family: des-cbc-crc, des-cbc-md5, and des-cbc-md4 (weak)
des3 - The triple DES family: des3-cbc-sha1
aes - The AES family: aes256-cts-hmac-sha1-96 and aes128-cts-hmac-sha1-96
rc4 - The RC4 family: arcfour-hmac
```

- Cryptographic Primitives
  - Cryptographic Agility (v5)
  - Etypes: *Define set of primitives for cryptographic operations*
    - e.g.: aes256-cts-hmac-sha1-96, aes128-cts-hmac-sha1-96, rc4-hmac, des-cbc-md5, rc4-hmac-exp

Hadoop Security & Services

HDFS: NN & DN

MapReduce: JT+TT

HBase: ZK+MASTER+RS

Pre-CDH3

- User Auth:
  - User impersonation: set property “hadoop.job.ugi” in run job
- Server Auth: N/A
- HDFS (weak-authentication)
  - Unix-like file permission (std: user/group/other r/w/x)
- Job control:
  - Lack of ACLs for counters/logging
  - ACLs per job queue submission/killing
- Web interface: N/A
- Tasks:
  - Not-isolated from the others
  - All run as same users
  - Interference with other tasks accessing identical local storage

Security ship w/ CDH3:

- Secure Authentication base on Kerberos
  - RPC secured with SASL GSSAPI mechanism
  - Strong authentication & SSO
- Mutual authentication between servers/users/services
  - Bi-directional for server auth.
- HDFS:
  - Same general permission model w/ sticky bit
- ACLs for job control & view
- Tasks isolation (launch by user) on same TT
- Kerberized SSL support for web interface (pluggable serverlet)

Authentication & Authorization

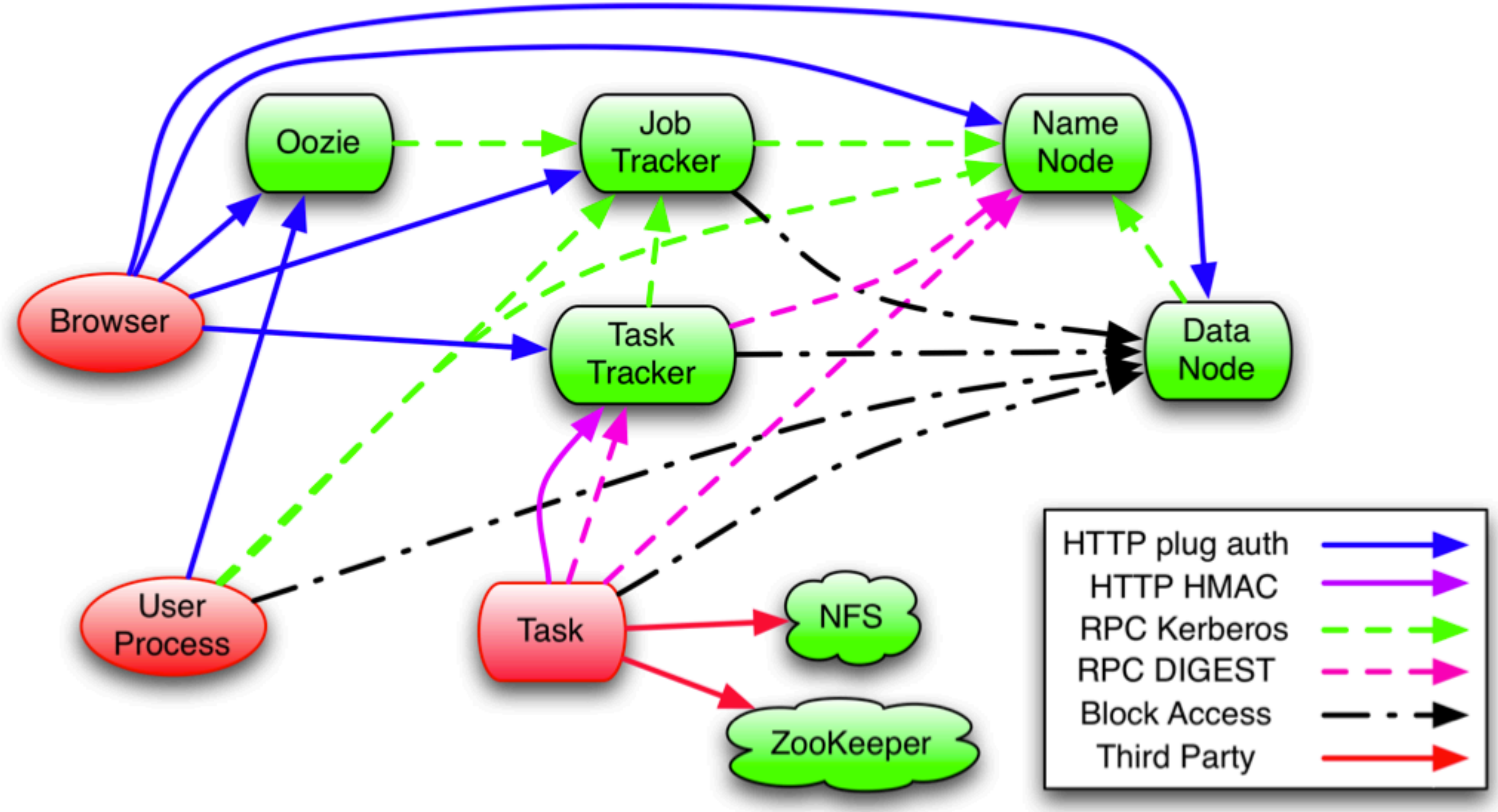
- Consideration
  - Performance: symmetric keys (Kerberos) vs. public key (SSL)
  - Management: central managed (KDC) vs. CRL propagation
- Authentication – *user identification*
  - Changes low-level transport
  - RPC authentication using SASL
    - Kerberos (GSSAPI)
    - Token (GIGEST-MD5)
    - Simple
  - HTTP secured via plugin
- Authorization – access control, resources & role
  - HDFS
    - Command line & semantics unchanged
    - Web UI enforces authentication
  - MapReduce added Access Control Lists
    - Lists of users and groups that have access
    - mapreduce.job.acl-view-job – view job
    - mapreduce.job.acl-modify-job – kill or modify job

etu
nexus enlightenment

Delegation Tokens

- To prevent a flood of authentication requests at the start of a job, NameNode can create delegation tokens.
- Allows user to authenticate once and pass credentials to all tasks of a job.
- JobTracker automatically renews tokens while job is running.
- Cancels tokens when job finishes.

Primary Communication Path



Ref: Hadoop Security Preview, Owen

Hadoop Security Enable

- In “core-site.xml”
  - Reset “simple” to disable security
  - Property:

  hadoop.security.authentication = kerberos

  hadoop.security.authorization = true

HDFS Security Configuration

- In "hdfs-site.xml", set property:

  dfs.block.access.token.enable = true

  dfs.namenode.keytab.file = *${HDFS_KEYTAB_PATH}*

  dfs.namenode.kerberos.principal = ${HDFS_KRB5_PRINCIPAL}

  e.g.: etu/_HOST@${HADOOP_REALM}

  dfs.namenode.kerberos.internal.spnego.principal =

  HTTP/_HOST@${HADOOP_REALM}

Secondary NN Configuration

- In "hdfs-site.xml", set the following property:
  - Similar properties as for NameNode
  - *Perfectly fine if you initiate with same Kerberos principal*

  dfs.secondary.namenode.keytab.file

  dfs.secondary.namenode.kerberos.principal

  dfs.secondary.namenode.kerberos.internal.spnego.principal

DataNode Security Configuration

- In “hdfs-site.xml”
- Replicate site xml to all DN
- Privilege service port:
  - *Either recompile “jsvc” or adopt BigTop for secure DN service daemon*
- “sudo” privilege required
- Appropriate variables for secured datanode

  HADOOP_SECURE_DN_USER
  HADOOP_SECURE_DN_PID_DIR (optional)
  HADOOP_SECURE_DN_LOG_DIR
  JSVC_HOME
- Define the following properties:

  dfs.datanode.data.dir.perm
  dfs.datanode.address *e.g.: 0.0.0.0:1004*
  dfs.datanode.http.address e.g.: 0.0.0.0:1006
  dfs.datanode.keytab.file
  dfs.datanode.kerberos.principal *e.g.: hdfs/_HOST@${HADOOP_REALM}*

Secure HDFS Service Common Error

- **Error:**

```
ERROR security.UserGroupInformation: PriviledgedActionException
as:etu (auth:KERBEROS) cause:javax.security.sasl.SaslException: GSS initiate failed
[Caused by GSSException: No valid credentials provided (Mechanism level:
Failed to find any Kerberos tgt)]

WARN ipc.Client: Exception encountered while connecting to the server :
javax.security.sasl.SaslException: GSS initiate failed [Caused by GSSException:
No valid credentials provided (Mechanism level: Failed to find any Kerberos tgt)]
12/10/02 11:03:24 ERROR security.UserGroupInformation: PriviledgedActionException
as:etu (auth:KERBEROS) cause:java.io.IOException: javax.security.sasl.SaslException:
GSS initiate failed [Caused by GSSException: No valid credentials provided (Mechanism level:
Failed to find any Kerberos tgt)]
```

- **C.f.: Successful Kerberos Authentication:**

```
Oct 02 11:06:16 master krb5kdc[30142](info): TGS_REQ (6 etypes {17 17 23 16 3 1})
10.1.247.18: ISSUE: authtime 1349147029, etypes {rep=17 tkt=17 ses=17},
etu@ETU.SYSTEX.TW for etu/master.etu.systex.tw@ETU.SYSTEX.TW
```

Secure MapReduce Configuration

- In “mapred-site.xml”, for JT & TT
  - Defined the following properties:

mapreduce.jobtracker.kerberos.principal

e.g.: mapred/_HOST@{HADOOP_REALM}

mapreduce.jobtracker.keytab.file

mapreduce.tasktracker.kerberos.principal

mapreduce.tasktracker.keytab.file

Secure MapReduce: TaskController

- In “mapred-site.xml”
- In taskcontroller.cfg:
  - Default “banned.users” property is mapred, hdfs, and bin
  - Default “min.user.id property” is 1000 (Err code: 255 if lower)
- Take care also ownership & setuid for taskcontroller binary
  - *chown root:mapred task-controller*
  - *chmod 4754 task-controller*
- Define also the following variables:

mapred.task.tracker.task-controller

e.g.: org.apache.hadoop.mapred.LinuxTaskController

mapreduce.tasktracker.group

e.g.: mapred

Secure MapReduce: Best Practice

- Always start with simple task before launch real workload: e.g.: PiEst
- Make sure underlying HDFS enable security & functional
- From KDC log:

```
master krb5kdc[30142](info): TGS_REQ (6 etypes {17 17 23 16 3 1})
192.168.70.18: ISSUE: authtime 1349147401, etypes {rep=17 tkt=17
ses=17},
etu@ETU.SYSTEX.TW for etu/master.etu.systex.tw@ETU.SYSTEX.TW
```

Zookeeper Security Configuration (I)

- **zoo.cfg:**

```
authProvider.
1=org.apache.zookeeper.server.auth.SASLAuthenticationProvider
jaasLoginRenew=3600000
```

- **java.env**

```
export JVMFLAGS="-Djava.security.auth.login.config=/etc/
zookeeper/conf/jaas.conf”
```

Zookeeper Security Configuration (II)

- **JAAS configuration:**

Server:

```
com.sun.security.auth.module.Krb5LoginModule required
 useKeyTab=true
 keyTab="/etc/zookeeper/conf/zookeeper.keytab"
 storeKey=true
 useTicketCache=false
 principal="zookeeper/fully.qualified.domain.name@<YOUR-REALM>"
```

Client:

```
com.sun.security.auth.module.Krb5LoginModule required
 useKeyTab=false
 principal="zkcli"
 useTicketCache=true
 debug=true
```

HBase Security Configuration

- **Authentication**
  - Identification mechanism for HBase servers & clients for HDFS, ZK & MR.
- **Authorization**
  - Ontop of coprocessor framework (AccessController): ACLs & allowable resources base on requesting users' identity
- **Configuration:**
  - Secure HBase servers: master & regionserver
  - REST API secure mode
  - JAAS configuration for secure ZK quorum servers
  - ACLs Configuration (table & column level)
    - grant, revoke, alter and permission display

etu

nexus enlightenment

HBase Severs w/ Secure HDFS Cluster

- Required by all HBase severs, both Master & RS (hbase-site.xml)
- Define following properties:

```
hbase.security.authentication
e.g.: kerberos

hbase.rpc.engine
e.g.: org.apache.hadoop.hbase.ipc.SecureRpcEngine

hbase.regionserver.kerberos.principal
e.g.: hbase/_HOST@${HADOOP_REALM}

hbase.regionserver.keytab.file
hbase.master.kerberos.principal
hbase.master.keytab.file
```

HBase: Secure ZK Quorum Connection

hbase-env.sh:

```
export HBASE_OPTS="$HBASE_OPTS -Djava.security.auth.login.config=/etc/hbase/conf/zk-jaas.conf"
export HBASE_MANAGES_ZK=false

kerberos.removeHostFromPrincipal=true
kerberos.removeRealmFromPrincipal=true
```

ZK JAAS configuration:

```
com.sun.security.auth.module.Krb5LoginModule required
useKeyTab=true
useTicketCache=false
keyTab="/etc/hbase/conf/keytab.krb5”
principal="hbase/fully.qualified.domain.name@<YOUR-REALM>";
```

HBase site xml, define also the following properties:

```
hbase.zookeeper.quorum = $ZK_NODES
hbase.cluster.distributed = true
```

etu
nexus enlightenment

HBase Authorization Configuration

- Required by all HBase severs, both Master & RS (hbase-site.xml)

hbase.security.authorization (true)

hbase.coprocessor.master.classes

e.g.: org.apache.hadoop.hbase.security.access.AccessController

hbase.coprocessor.region.classes

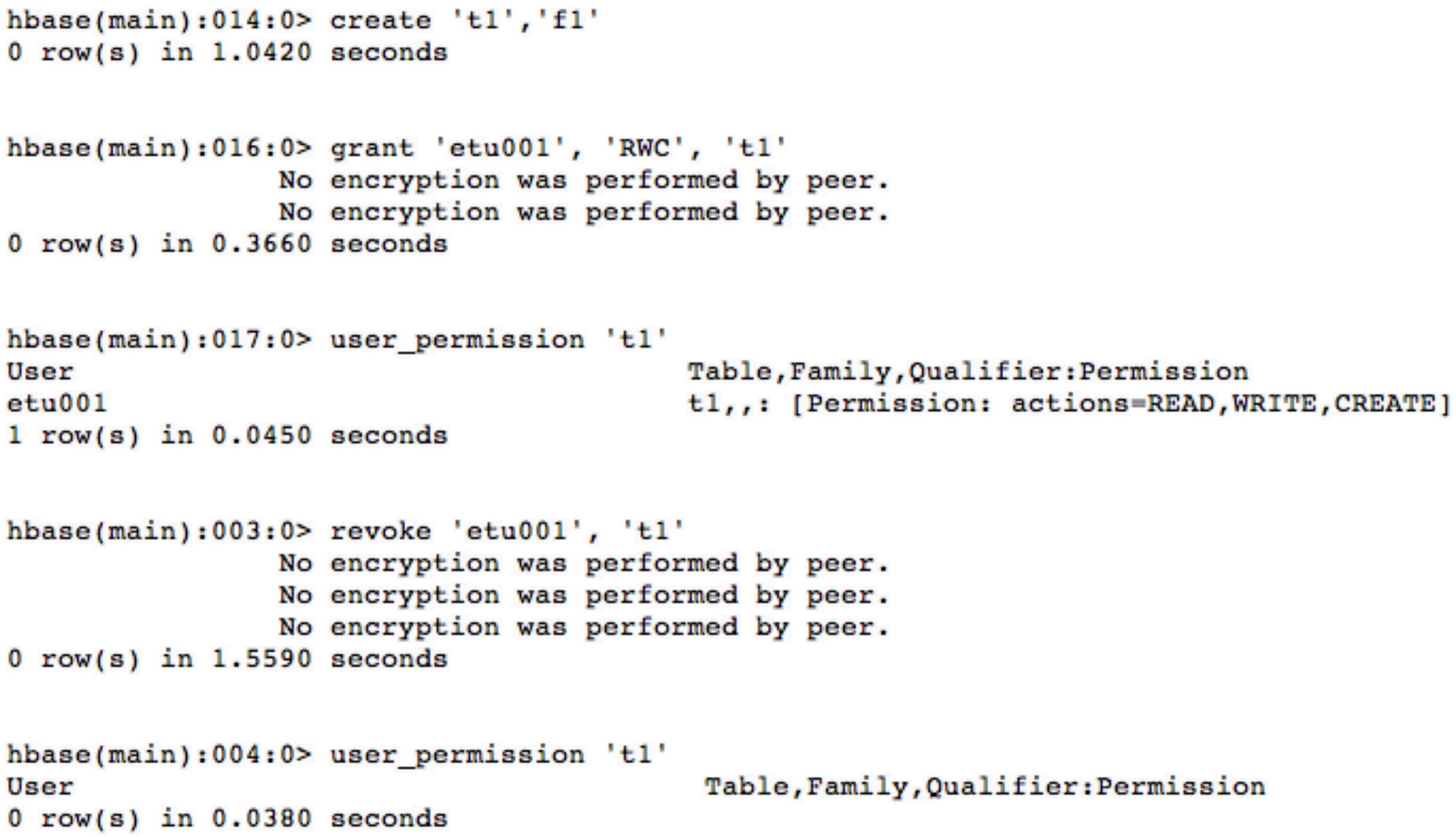
e.g.: org.apache.hadoop.hbase.security.token.TokenProvider, org.apache.hadoop.hbase.security.access.AccessController

HBase ACLs Rules

ACLs	Permissions
R/Read	Get, Scan, or Exists calls
W/Write	Put, Delete, LockRow, UnlockRow, IncrementColumnValue, CheckAndDelete, CheckAndPut, Flush, & Compact
C/Create	Create, Alter, & Drop
A/Admin	Enable, Disable, MajorCompact, Grant, Revoke, & Shutdown.

etu
nexus enlightenment

HBase: ACLs for Authorization

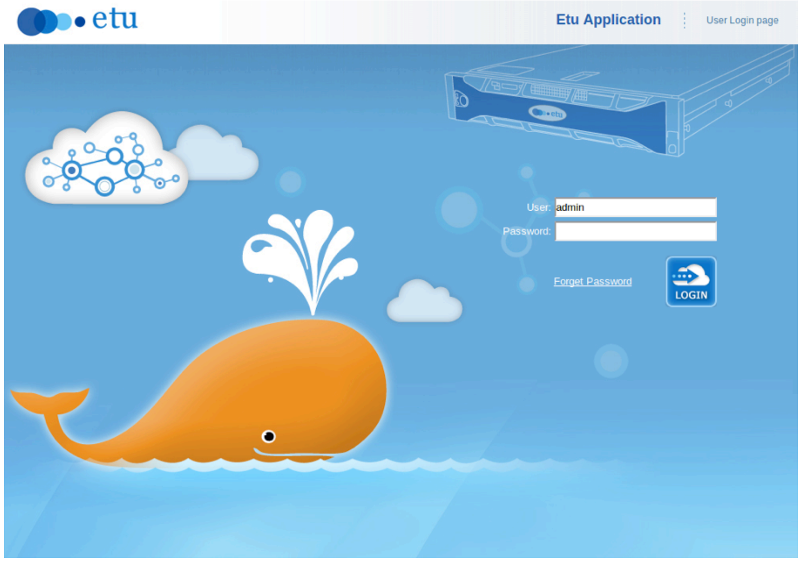


```
hbase(main):014:0> create 't1','f1'
0 row(s) in 1.0420 seconds


hbase(main):016:0> grant 'etu001', 'RWC', 't1'
                No encryption was performed by peer.
                No encryption was performed by peer.
0 row(s) in 0.3660 seconds


hbase(main):017:0> user_permission 't1'
User                                    Table,Family,Qualifier:Permission
etu001                                  t1,,: [Permission: actions=READ,WRITE,CREATE]
1 row(s) in 0.0450 seconds


hbase(main):003:0> revoke 'etu001', 't1'
                No encryption was performed by peer.
                No encryption was performed by peer.
                No encryption was performed by peer.
0 row(s) in 1.5590 seconds


hbase(main):004:0> user_permission 't1'
User                                     Table,Family,Qualifier:Permission
0 row(s) in 0.0380 seconds
```

Troubleshooting

- **Misconfiguration**?
  - Pseudo-distributed to cluster-wide configuration
  - Full cluster functionality before kerberizing services
  - Correct principal & keytab contains up-to-date KVNO.
  - Disentangle security related settings to isolate root causes
    - Ticket renewable fail? or expired.
- **System-wide**
  - Permission (files, directories and ownership), objClasses & ACLs
  - System clock screw, KDC operation (REALM), file handle limitation? (ulimit)
  - TT, RS, DN fail to start? Out of disk space? "dfs.datanode.du.reserved"
  - Name resolve (reverse), routing (multi-channels) etc.
- **Extensive debugging info**
  - Increase root.logger level, e.g.: hadoop.root.logger & hadoop.security.logger
  - Security mode: "-Djavax.net.debug=ssl -Dsun.security.krb5.debug=true"
- **Correct Hadoop "home" to look into?**
- **Relevant logging system:**
  - KDC log provide: TGS & AS req., principals, authtime and etypes.

etu
nexus enlightenment



Etu Appliance

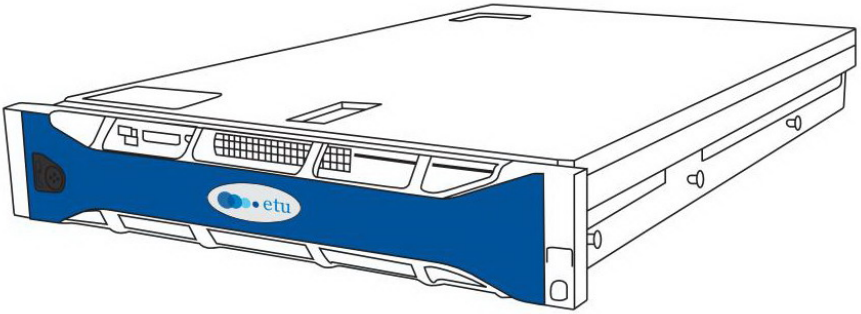
New features & key benefits

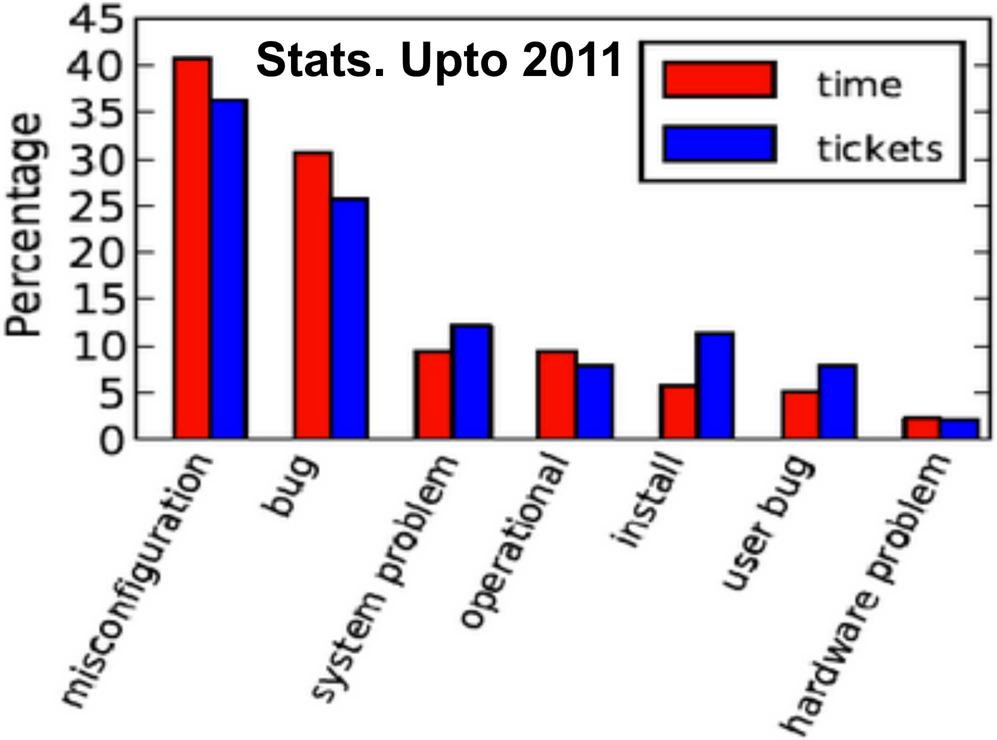
Software stacks & versions

HW spec.

Why appliance?

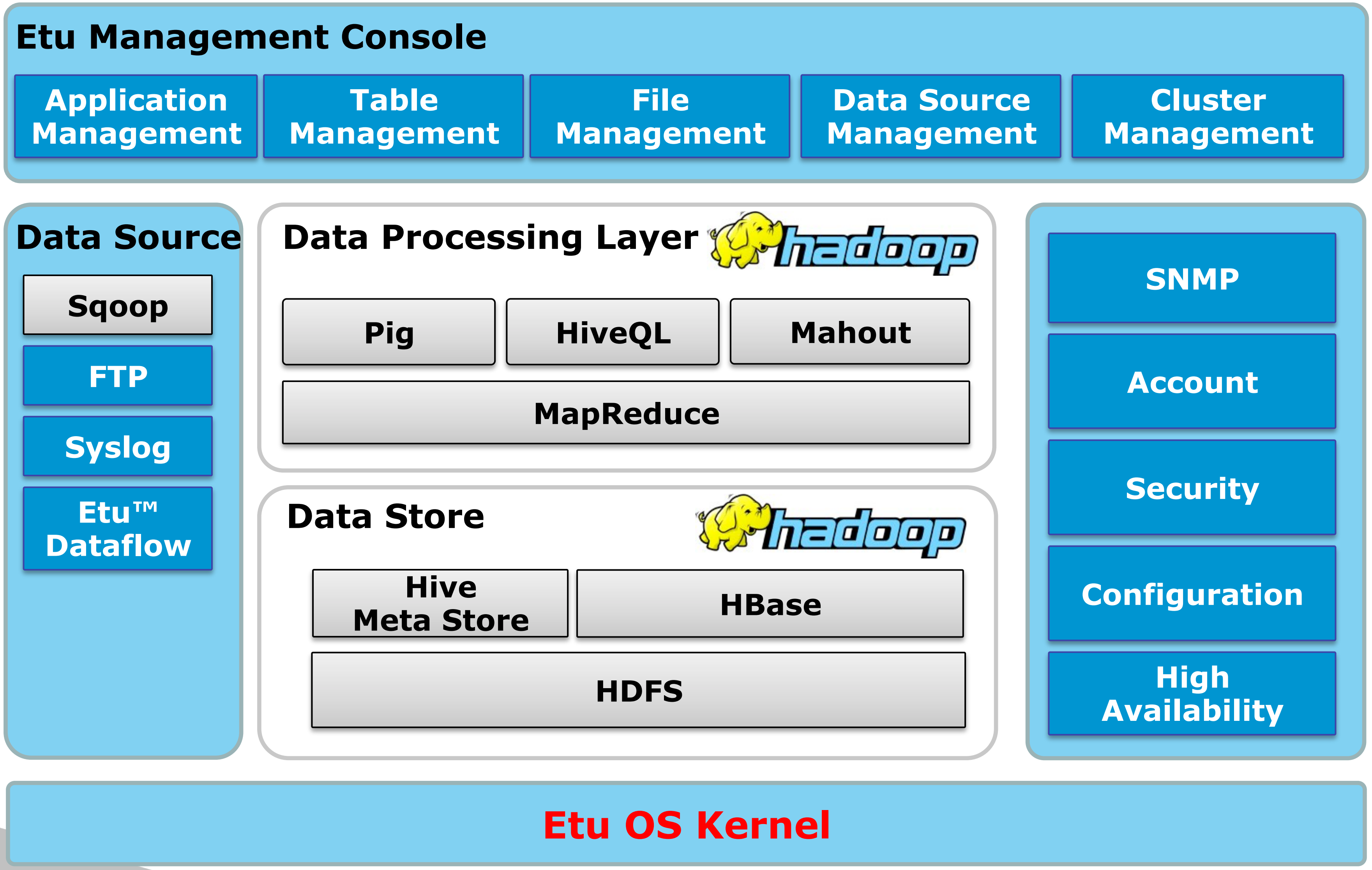
- Misconfiguration comprise 35% of tickets
  - Generic issues: memory allocation, disk spaces & file handling
- ~40% refer to system-wide and operation issues.
  - System automation, robust deployment, dashboard and event management strictly required for production operation





Ref: Hadoop Troubleshooting, Kate, Cloudera

Software Stack



Etu Management Console

- Application Management
- Table Management
- File Management
- Data Source Management
- Cluster Management

Data Source

- Sqoop
- FTP
- Syslog
- Etu™ Dataflow

Data Processing Layer

- Pig
- HiveQL
- Mahout
- MapReduce

Data Store

- Hive Meta Store
- HBase
- HDFS

- SNMP
- Account
- Security
- Configuration
- High Availability

Etu OS Kernel

etu
nexus enlightenment

Etu References:

- **Chiang, Fred**. (Deputy Vice President) **"Big Data 101 一一個充滿意圖與關聯世界的具體實現"** SYSTEX行雲流水系列(三), 24 May 2012. http://www.slideshare.net/fchiangtw/big-data-101
- **Chen, James.** (Principal Consultant of Etu) **"Hadoop 與 SQL 的甜蜜連結"** SYSTEX行雲流水系列(三), 24 May 2012. http://www.slideshare.net/chaoyu0513/hadoop-sql
- **Wu, Jeng-Hsueh.** (Principal Architect of Etu), **"Facing the Big Data challenge: a use case for leveraging from Hadoop and her friends",** OSDC, 14 Apr 2012. http://osdc.tw/schedule
- **Nien, Johnny.** (Technical Manager) **"Etu DataFlow: An efficient data streaming & pre-processing framework designed for Hadoop"**, COSCUP, 19 Aug 2012. http://coscup.org/2012/en/program

etu
nexus enlightenment

Hadoop Security References:

- **Cloudera**
  - CDH3 Security Guide
  - CDH4 Beta 2 Security Guide
- **Hadoop Security**
  - *Slideshare*
  - **"Hadoop Security Design"**, Owen O'Malley et. al., Oct 2009
  - **"Integrating Kerberos into Apache Hadoop"**, Owen O'Malley
  - **"Plugging the Holes: Security and Compatibility"**, Owen O'Malley
  - **"Developing and deploying Apache Hadop Security"** Hortonworks, Owen
  - **"Hadoop Security, Cloudera"** Hadoop World 2010, Todd Lipcon & Aaron Myers
- **Kerberos & LDAP**
  - Administration: http://web.mit.edu/Kerberos/krb5-1.8/krb5-1.8.3/doc/krb5-admin.html
  - Installation: http://web.mit.edu/Kerberos/krb5-1.8/krb5-1.8.3/doc/krb5-install.html
  - Openldap: http://www.openldap.org/doc/admin24/dbtools.html

nexus enlightenment etu

Question?

jasonshih@etusolution.com

nexus enlightenment etu

Contact

www.etusolution.com
info@etusolution.com

Taipei, Taiwan
318, Rueiguang Rd., Taipei 114, Taiwan
T: +886 2 7720 1888
F: +886 2 8798 6069

Beijing, China
Room B-26, Landgent Center,
No. 24, East Third Ring Middle Rd.,
Beijing, China 100022
T: +86 10 8441 7988
F: +86 10 8441 7227