Metro 2.0 – the advanced SOAP web services stack in GlassFish

metro.dev.java.net

Harold Carr
Architect, Project Metro

Metro – SOAP-based Web Services Stack

- Built into: GlassFish, Java CAPS, OpenSSO
- Works with any servlet 2.5 compliant web container
  - > WebLogic, WebSphere, JBoss, Tomcat, …
  - > Also standalone
- Advanced Interoperability with .NET 3.x/4.0
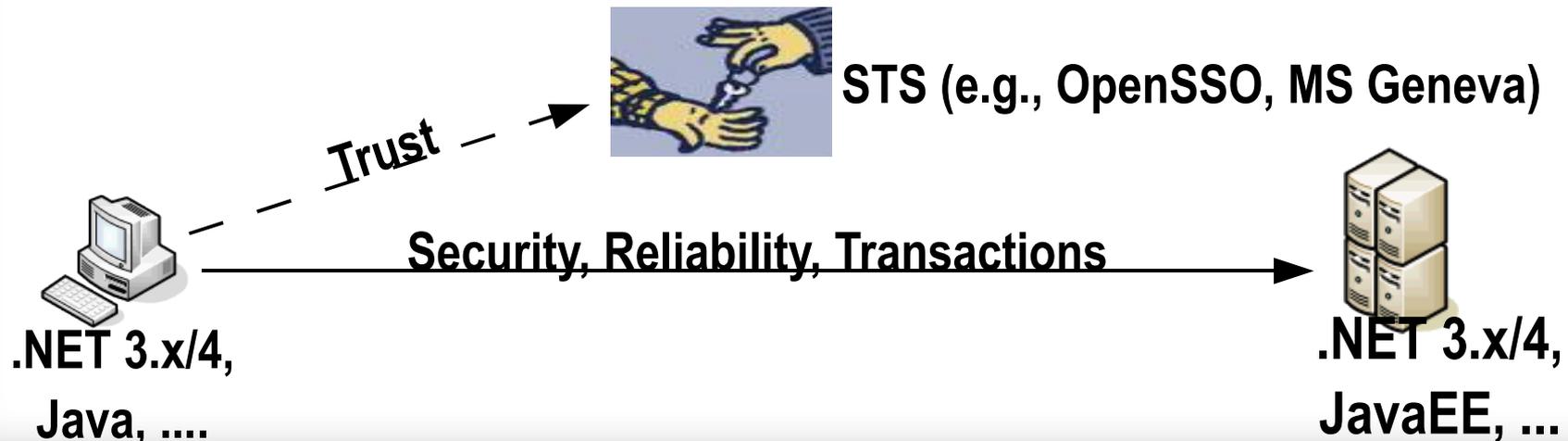
STS (e.g., OpenSSO, MS Geneva)

Trust

Security, Reliability, Transactions

.NET 3.x/4,

Java, ...

.NET 3.x/4,

JavaEE, ...

Metro: main features

- JAX-WS 2.2 (new version)
  > autogen JAXB beans (new)
- Security
  > password derived keys (new)
  > @RolesAllowed (new)
- Identity
  > SSO for services (new)
  > Identity delegation (new)
- Reliable Messaging
  > persistent (new)

- Atomic Transactions
- Centralized config (new)
- Monitoring (new)
- NetBeans support
- Eclipse support (new)
- Extensible
- Standards-based
  > addressing metadata (new)
  > WS-MakeConnection (new)

Metro Interoperability
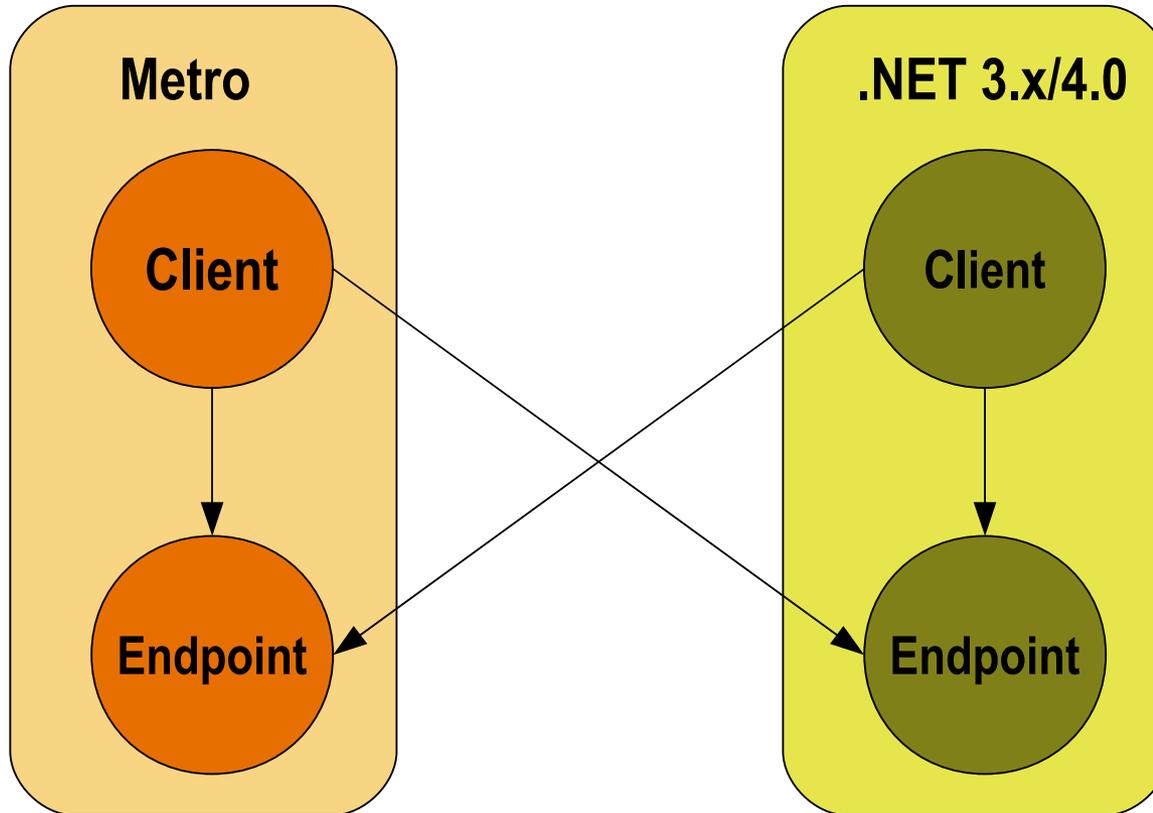
Metro SOAP Web Services Stack

JAX-WS Tooling, NetBeans & Studio Support Software

Security
Secure Conv.
Trust
XWSS

Reliability
Reliable-
Messaging

Transactions
Atomic-
Transactions
Coordination

Metadata
WSDL
MEX
Policy

SOAP Based Messaging (WSA, MTOM)

JAXB Based XML Data Binding (XSD, XPATH)

HTTP   TCP   SMTP

# Metro : Security

# Metro : clear text communication
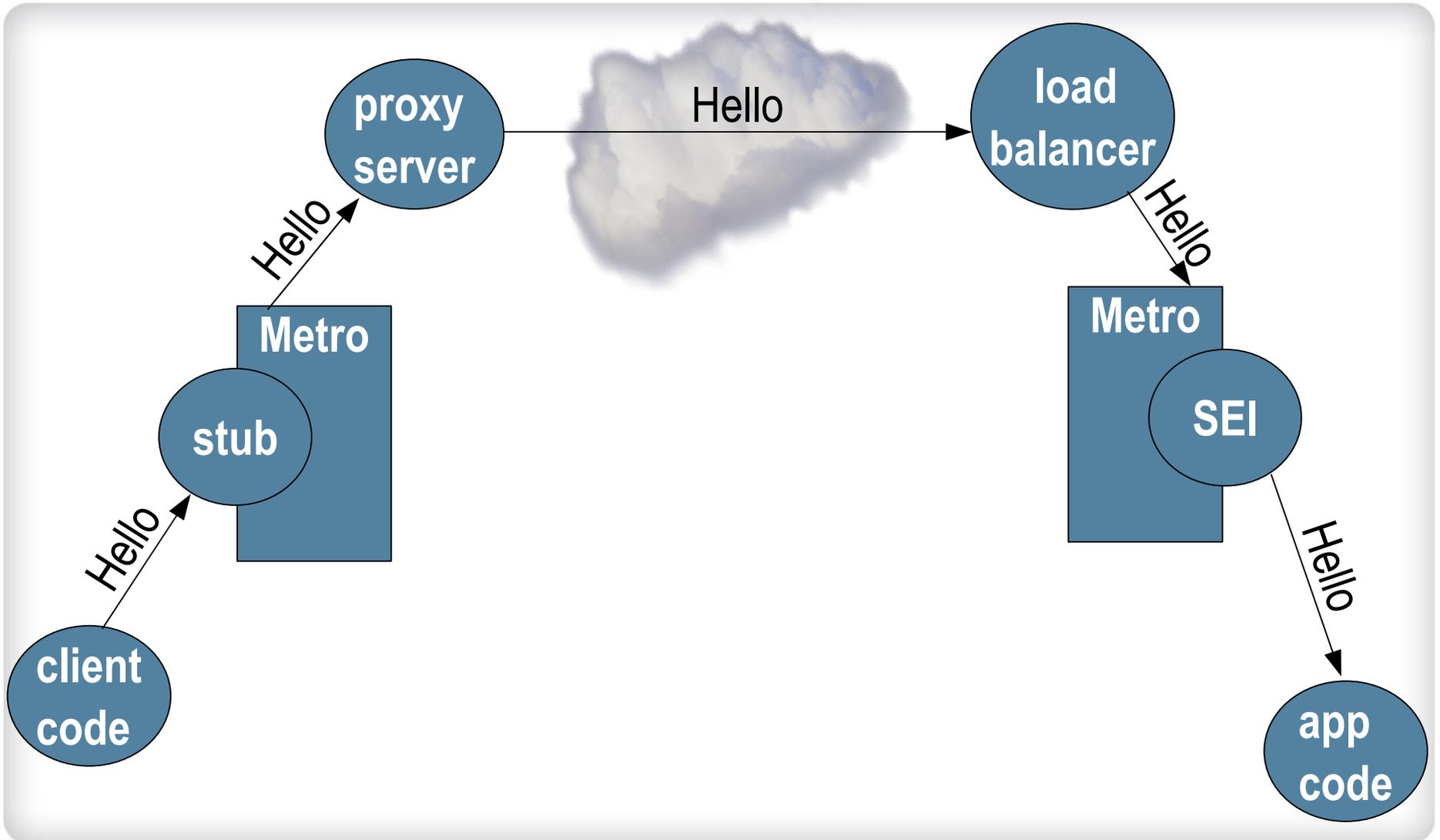
Metro : HTTPS communication
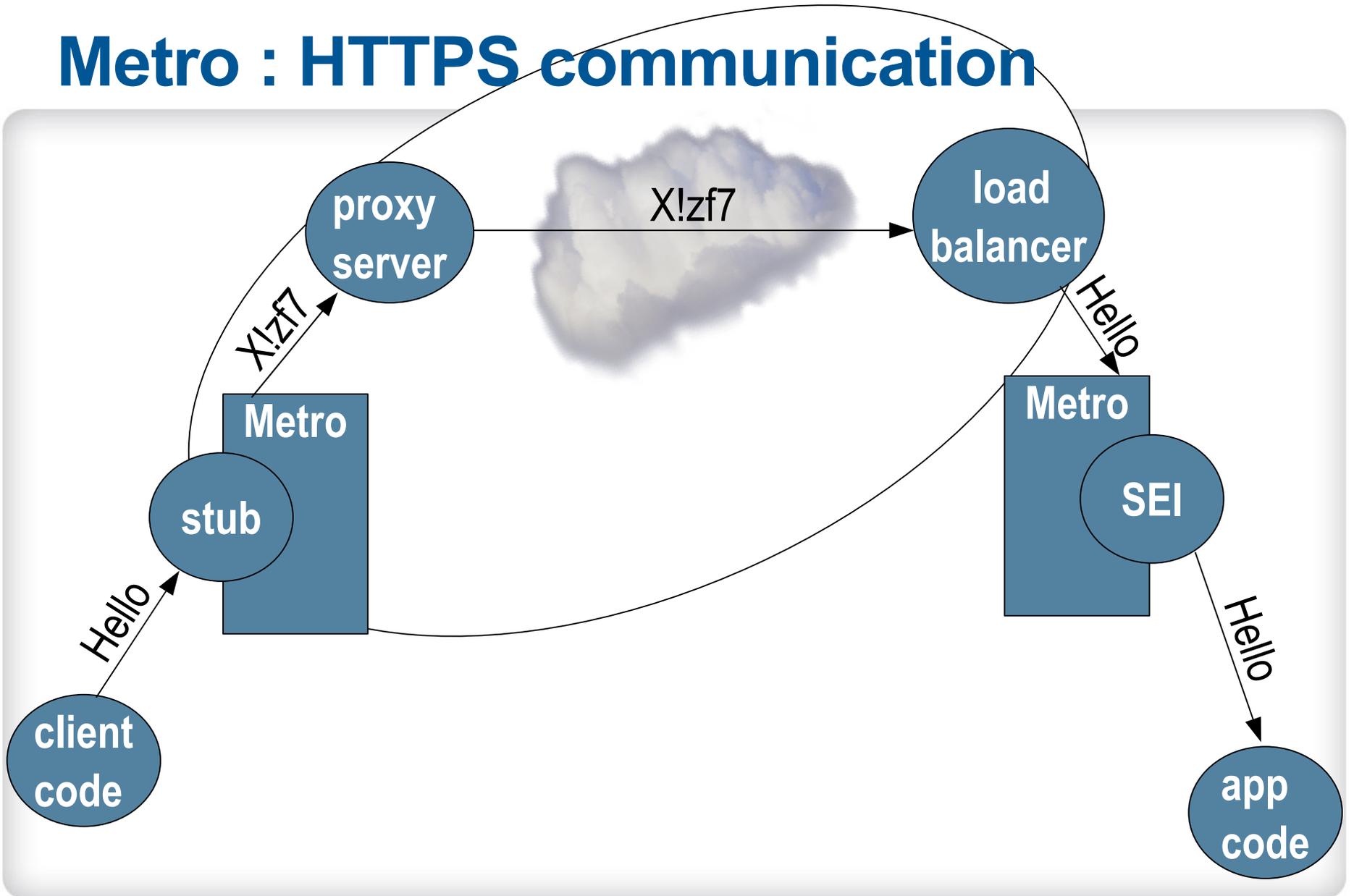
Metro : end-to-end secure comm
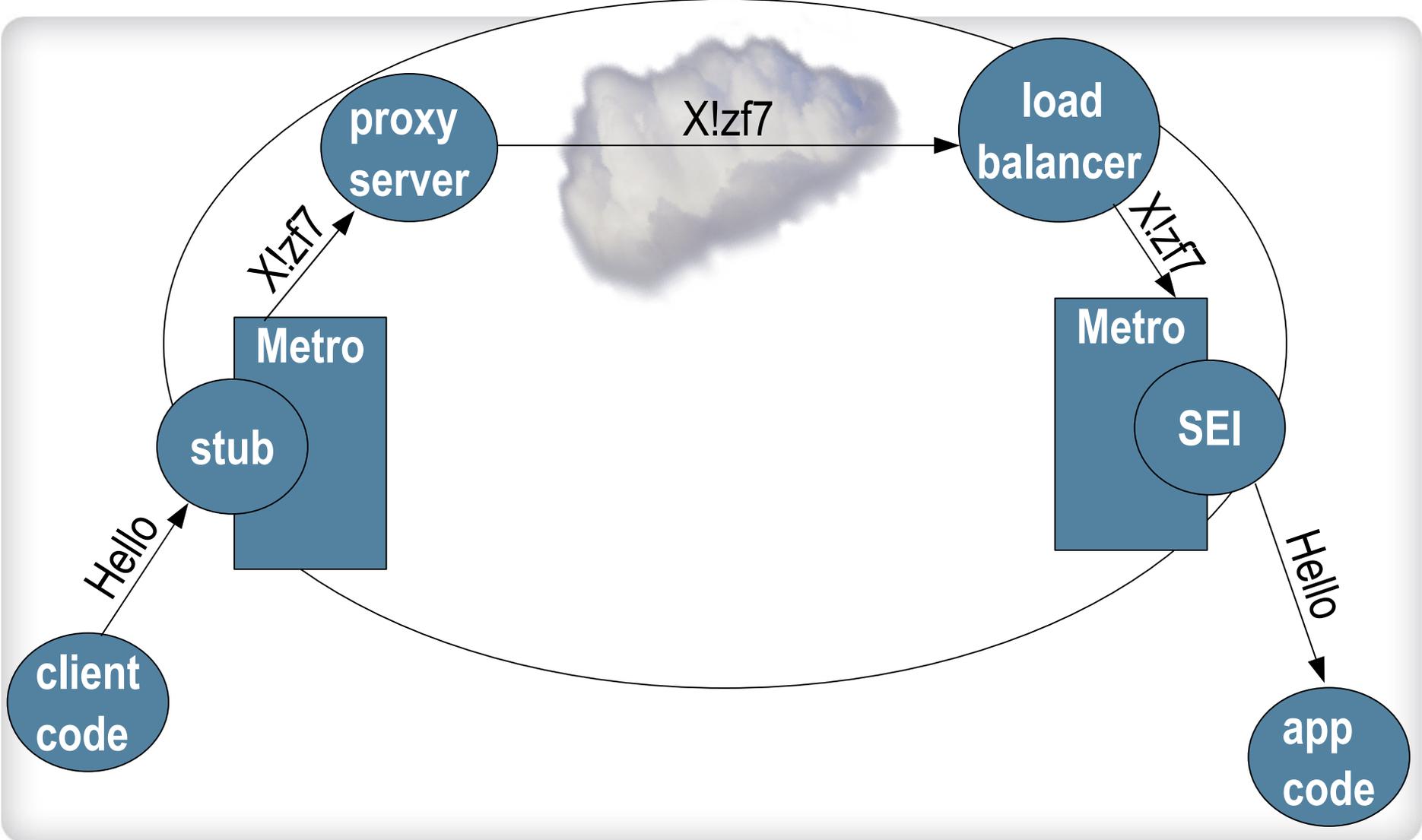
Metro: mutual certificates security
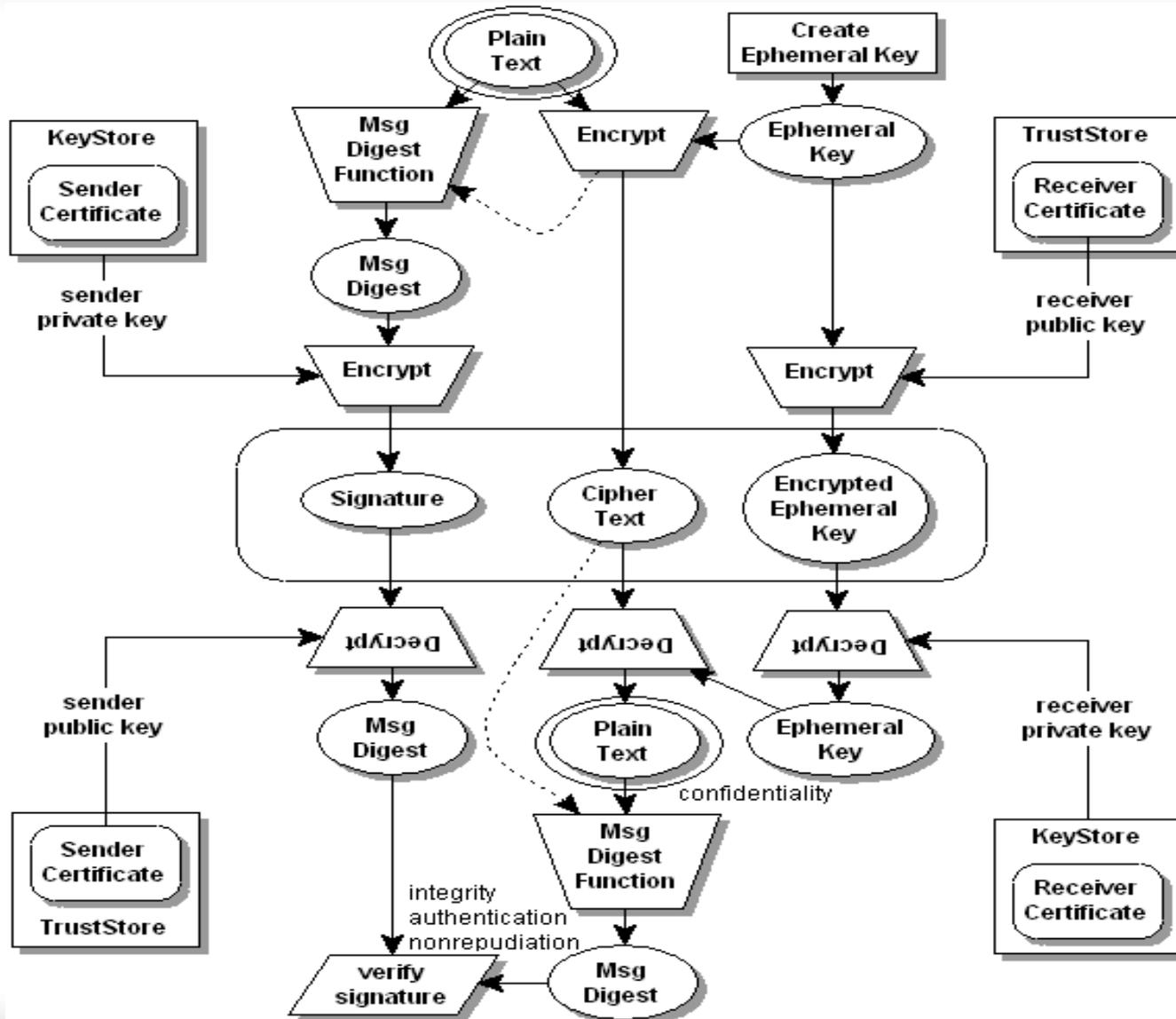
Metro: creating a web service

```java
package foo;

import javax.jws.WebService;

@WebService()
public class MyService {

    public String echo(final String x) {
        return "echo: " + x;
    }
}
```

Metro: adding security using NetBeans

ServicePortBinding

Version Compatibility: .NET 3.5 / METRO 1.3 (requires METRO 1.3 or higher)

☐ Optimize Transfer Of Binary Data (MTOM)

☐ Reliable Message Delivery

☐ Deliver Messages In Exact Order

Advanced...

☑ Secure Service

Security Mechanism: Mutual Certificates Security

Mutual certificates for authentication and message integrity and confidentiality protection.

| ✓ Mutual Certificates Security |
| Symmetric Binding with Kerberos Tokens |
| Transport Security (SSL) |
| Message Authentication over SSL |
| SAML Authorization over SSL |
| Endorsing Certificate |
| SAML Sender Vouches with Certificates |
| SAML Holder of Key |

☐ Use Development Defaults

Keystore... Truststore... Kerberos...

Validators... Advanced...

Option: Secure Conversation
Use: increase performance/security on multiple msgs

WSS vs WSSC vs TLS

Legend:
- WSS
- WSSC(with KD)
- WSSC(without KD)
- TLS

KD = Key Derivation

y-axis: time in milliseconds (0, 2000, 4000, 6000, 8000)
x-axis: number of messages (1, 2, 5, 10)

Option: Secure Conversation
Use: increase performance/security on multiple msgs

WSS vs WSSC vs TLS

Legend:
- WSS
- WSSC(with KD)
- WSSC(without KD)
- TLS

KD = Key Derivation

time in milliseconds
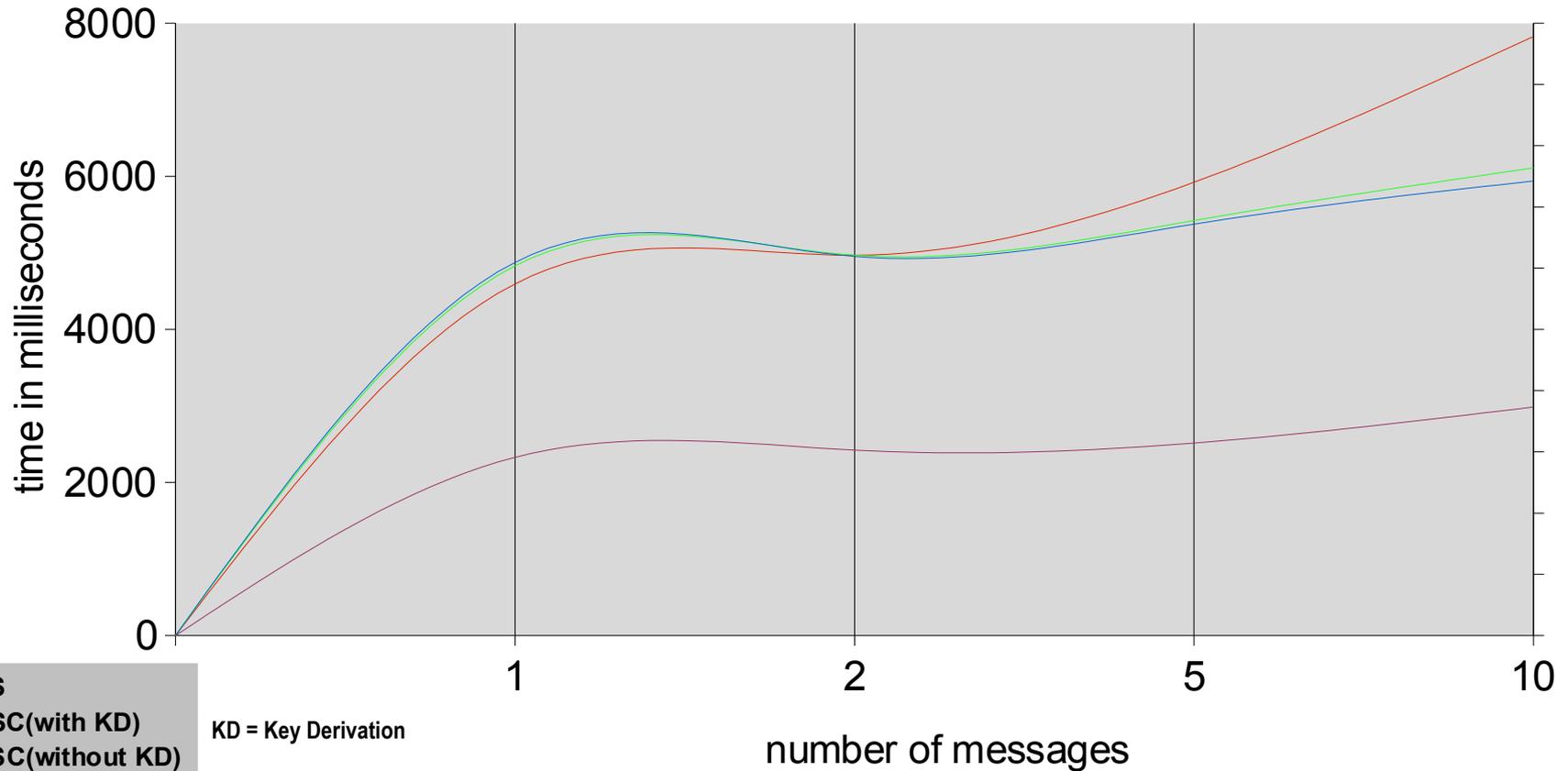
number of messages
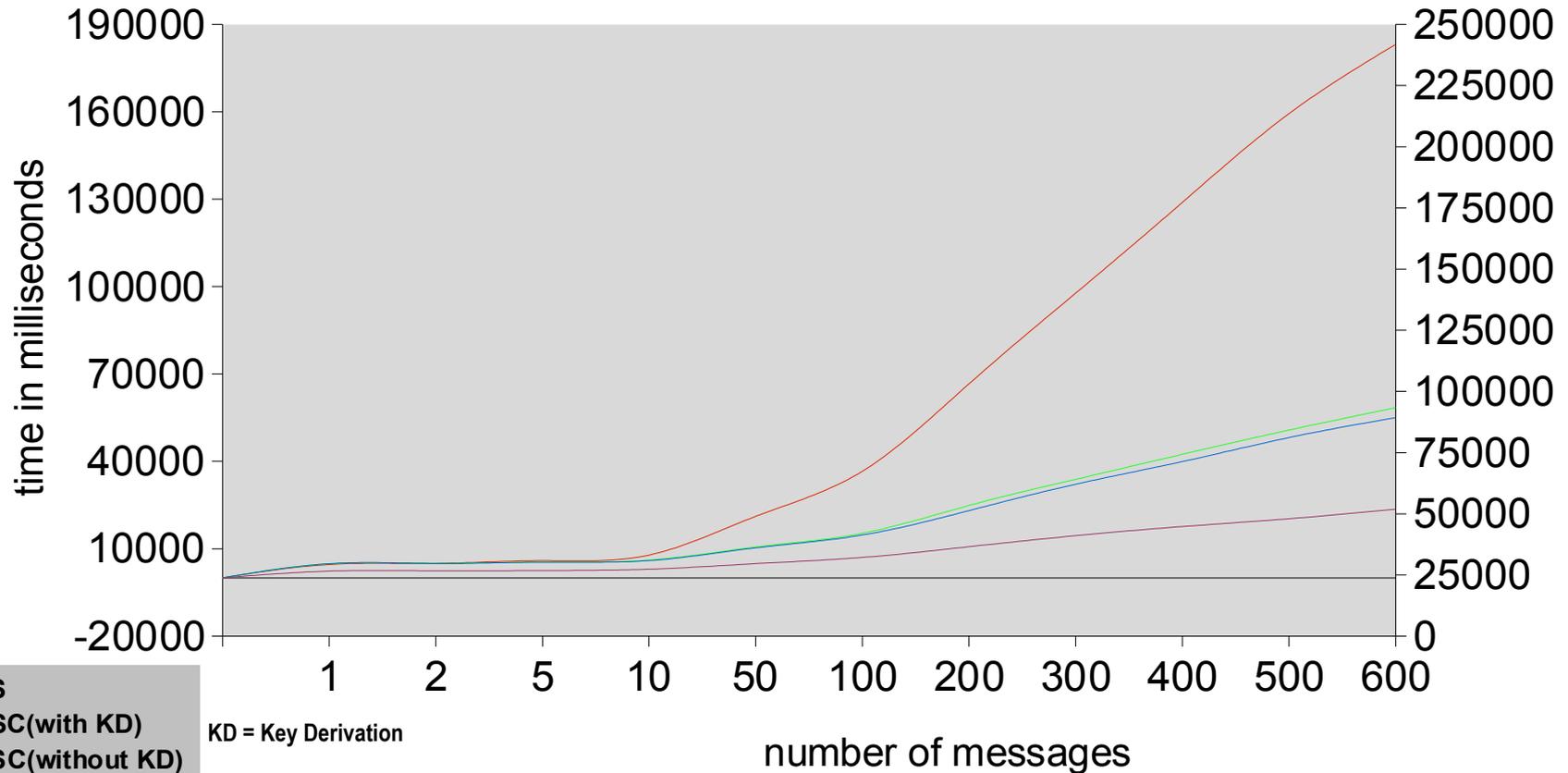
(note: baseline is plain text)

Metro : Identity

Metro : authentication
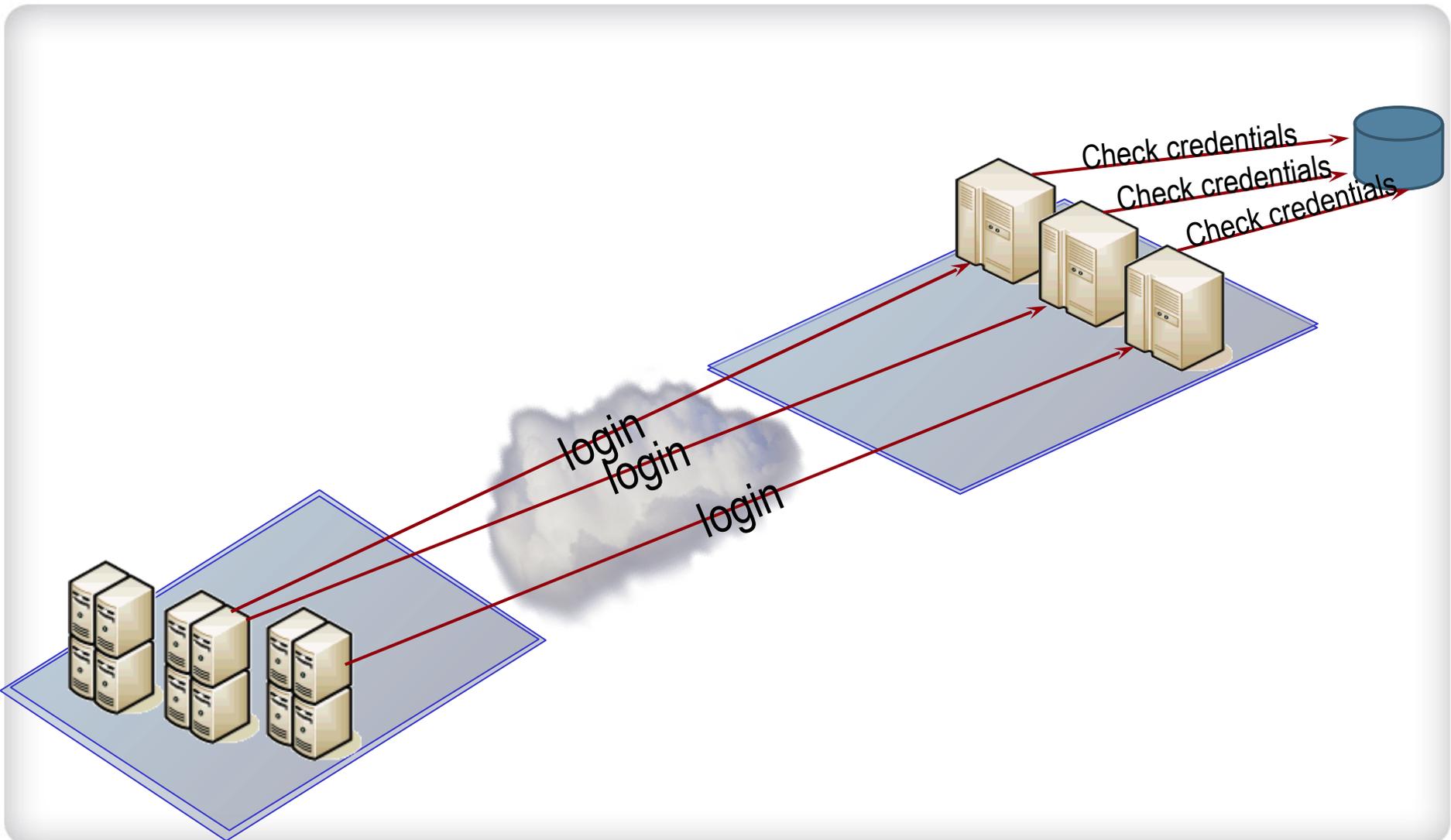
Metro : 3rd party authentication

Secure Token Service (STS)
e.g., OpenSSO

Check credentials

login
login

msg (with token)
msg (with token)
msg (with token)
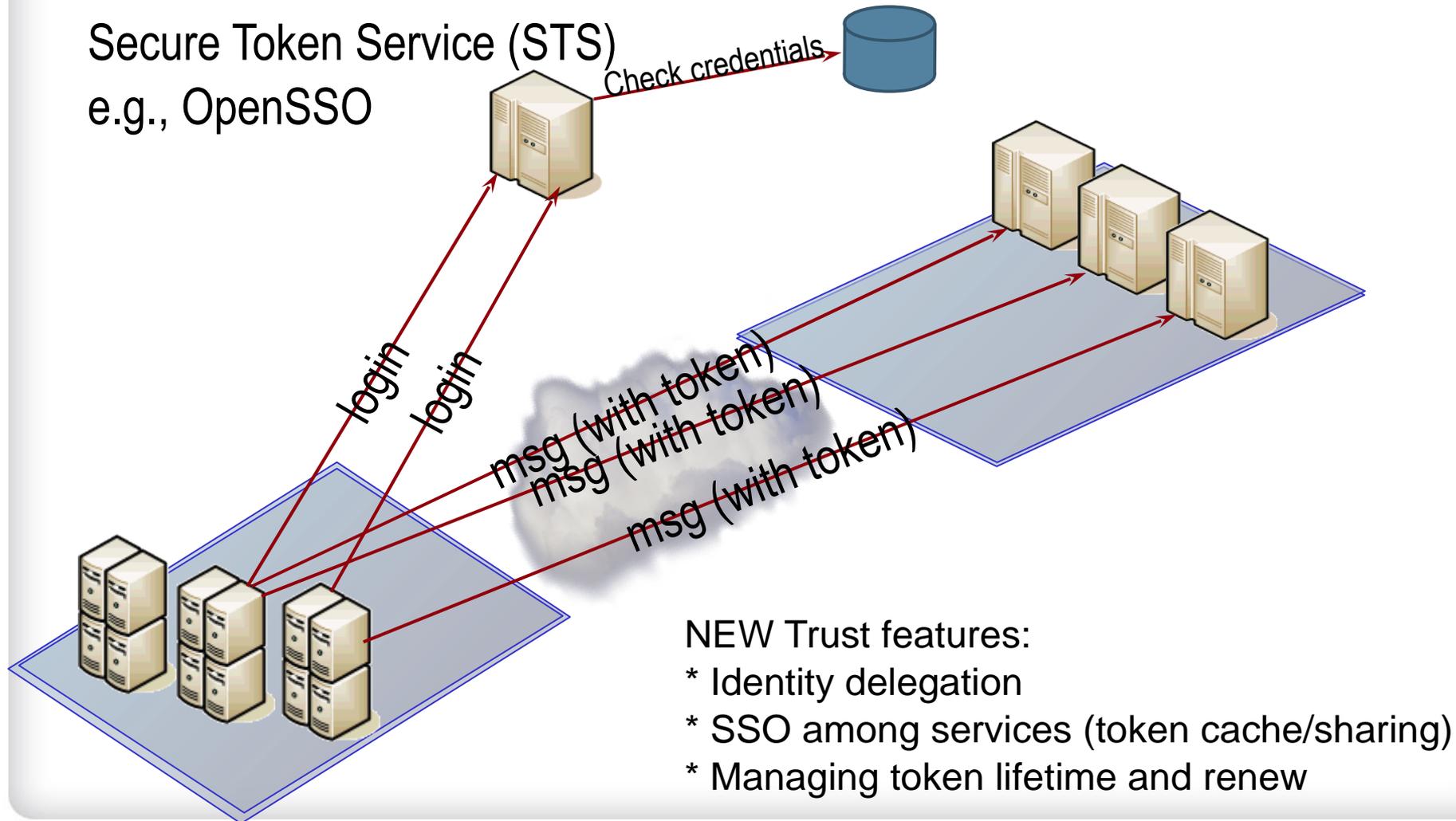
NEW Trust features:
* Identity delegation
* SSO among services (token cache/sharing)
* Managing token lifetime and renew

Metro : brokered trust
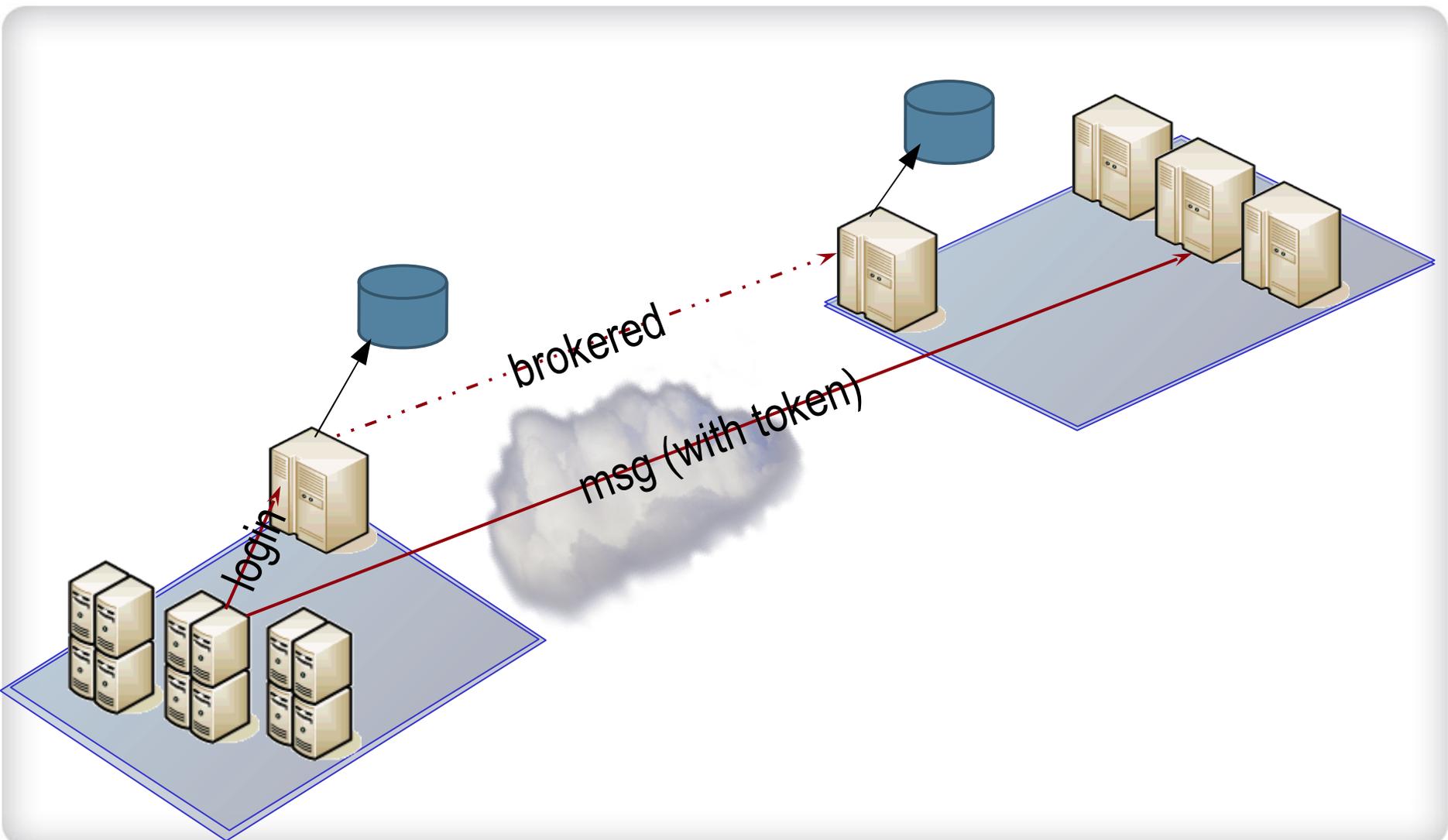
Metro : Reliable Messaging

Metro : Reliable Messaging

Recovery from
- missing messages
- out-of-order messages
- server-side crash/restart (NEW)
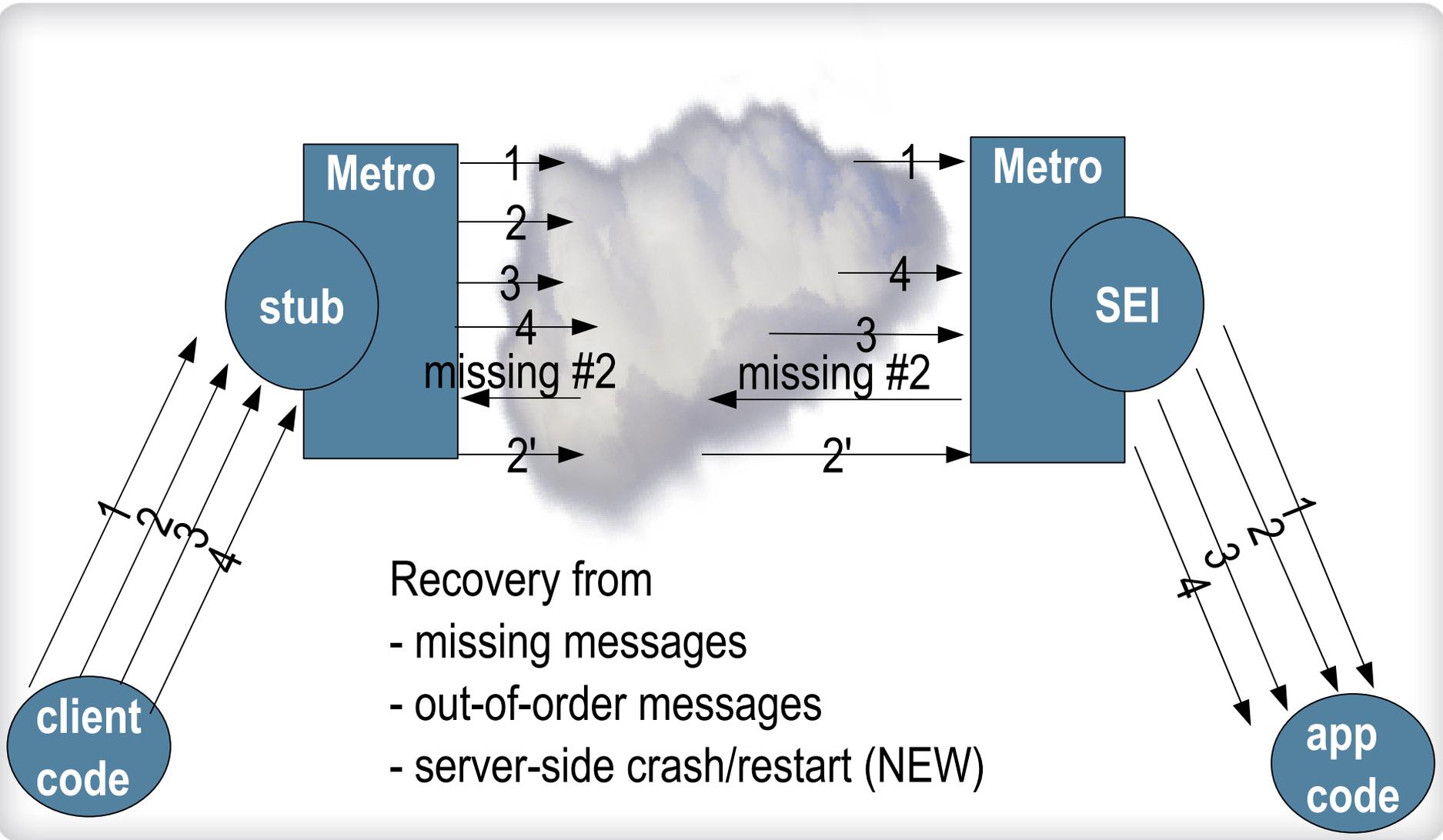
Metro : Atomic Transactions

Metro : Atomic Transactions

```
UserTransaction ut = context.getUserTransaction();

try {
    ut.begin();
    accountA.withdraw(amount);
    accountB.deposit(amount);
    ut.commit();
} catch (Exception ex) {
        ut.rollback();
}
```

all or nothing

Metro : Monitoring & Management

Metro : Monitoring & Management

Metro Management using jmxterm

```
$>open localhost:8686
```

```
$>beans
amx:name=/RMService-RMServiceService-RMServicePort,pp=...


$>bean amx:name=/RMService-RMServiceService-RMServiceP...


$>get features
features = [ {
  enabled = true;
  iD = com.sun.xml.ws.rm.ReliableMessagingFeature;
}, {
  enabled = true;
  iD = http://www.w3.org/2005/08/addressing/module;
} ];


$>set dumpHTTPMessages true
```

Metro : centralized web service configuration management
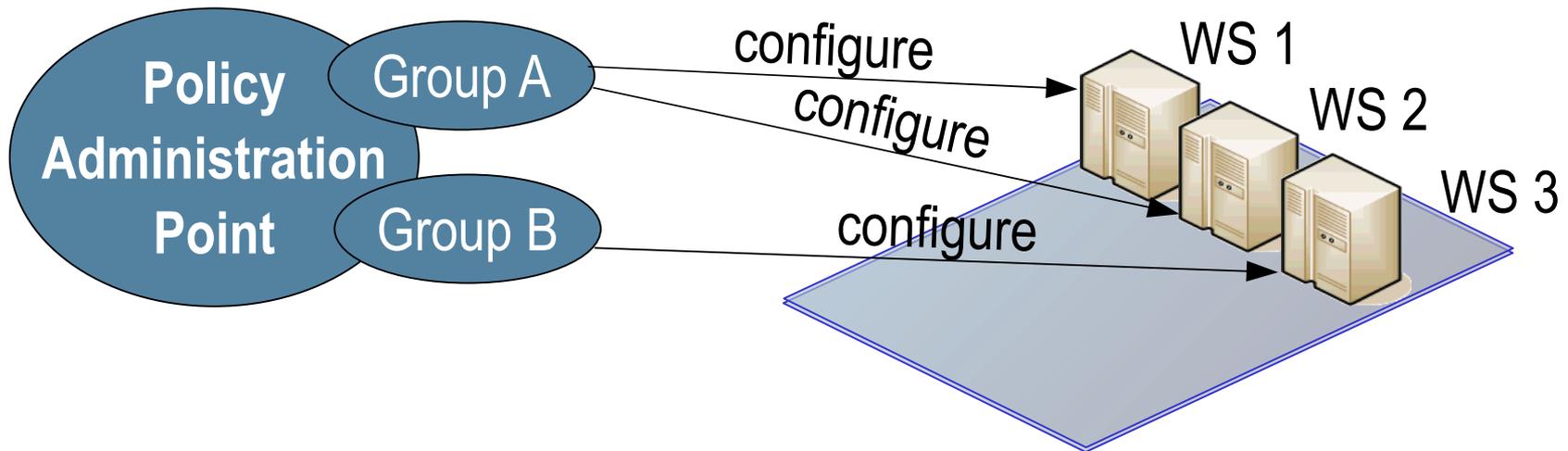
Metro Interoperability via Standards

- Basic Web Services
  - > JCP: JAX-WS 2.2 & JAXB 2.2
  - > W3C: SOAP 1.1/1.2, WSDL 1.1, WS-Addressing, MTOM
  - > WS-I: Basic Profile 1.x, 2.0, SSBP 1.0, AP 1.0, BSP 1.0
- Enterprise Web Services
  - > Oasis: WS-Security, WS-SecureConversation, WS-Trust, WS-SecurityPolicy, WS-ReliableMessaging, WS-MakeConnection, WS-AtomicTransactions, WS-Coordintation
  - > W3C: WS-Addressing, WS-Policy, WS-Transfer
  - > WS-MetadataExchange
- (Same specs implemented by MS .NET)

Metro: main features

- JAX-WS 2.2 (new version)
  - > autogen JAXB beans (new)
- Security
  - > password derived keys (new)
  - > @RolesAllowed (new)
- Identity
  - > SSO for services (new)
  - > Identity delegation (new)
- Reliable Messaging
  - > persistent (new)

- Atomic Transactions
- Centralized config (new)
- Monitoring (new)
- NetBeans support
- Eclipse support (new)
- Extensible
- Standards-based
  - > addressing metadata (new)
  - > WS-MakeConnection (new)
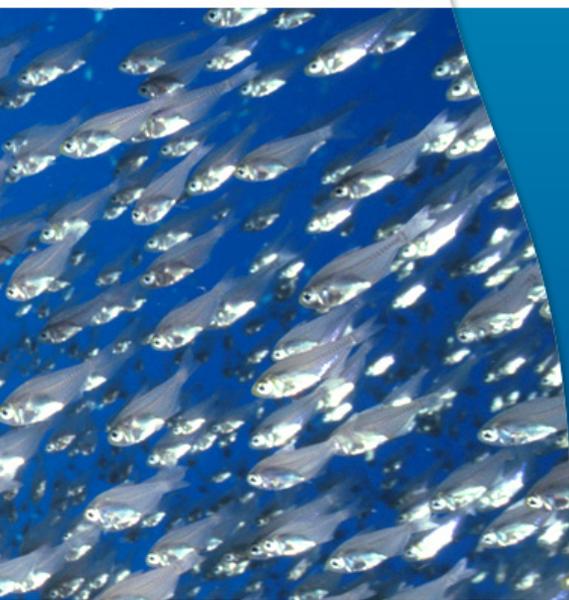
Metro SOAP Web Services stack in GlassFish v3

metro.dev.java.net