ISACA
In dienst van IT Governance-specialisten
Netherlands Chapter

1secure

SOA Security

en de rol van de auditor...

ISACA Roundtable
2 juni 2008

Arthur Donkers, 1Secure BV
arthur@1secure.nl

SOA

“Web 2.0, web services en service oriented architecture (SOA) is tegenwoordig de manier om de schat aan (legacy) gegevens die binnen organisaties aanwezig is te ontsluiten naar verschillende applicaties.”

SOA Security

Maar gebeurt dat op een veilige en controleerbare manier ?

En hoe controleer je dat eigenlijk?

SOA Security Praktijkervaringen

Tijdens het ontwerp van een security architectuur voor een Enterprise Service Bus bij een klant is een aantal praktijk-ervaringen opgedaan.

Deze presentatie vat een aantal van de belangrijkste ervaringen weer die van pas kunnen komen bij audits.

Geen panklaar werkprogramma (nog niet)

Agenda

Even voorstellen

Buzzword Bingo

Applicatie architectuur, toen en nu

Praktijkcase, Ontwerp en audit aspecten

Verbeterpunten

Belang van ontwikkelproces

Vragen

Meer weten ?

ir. Arthur Donkers

CISSP, CISA, CISM
Security Architecture
Management & Forensics

1secure
Burg. F. v. Ankenweg 5
9991 AM Middelstum

T +31 595 557057
F +31 595 557046
I www.1secure.nl
E arthur@1secure.nl

Wie ben ik ?

Arthur Donkers (CISSP, CISA, CISM)

Architect informatiebeveiliging

Security Management

Forensics

Auditor en Ethical Hacker

Balans tussen business en beveiliging

Wie is 1Secure ?

Onafhankelijk bedrijf voor informatie beveiliging

Advies, audit, architectuur en forensics

Nationaal en internationaal

Technische en organisatorische aandacht

Projecten bij bedrijfsleven en overheid

Nog even dit…

Vragen ?

Graag !

Buzzword Bingo

XML	eXtended Markup Language
SOAP	Simple Object Access Protocol
.NET	… MSmarketing …
WSDL	Web Service Definition Language
XAML	eXtensible Application Markup Language
AJAX	Asynchronous Javascript and XML

Buzzword Bingo

SAML(2)	Security Assertion Markup Language
XMLRPC	Remote Procedure Calls over XML
SOA	Service Oriented Architecture
XKMS	XML Key Management Specification
WS Security	Uitbreiding op SOAP voor berichten uitwisseling
BPEL	Business Process Engineering Language

Referenties

http://www.w3schools.com/soap/soap_intro.asp

http://www.w3.org/TR/wsdl

http://www.w3.org/TR/xkms/

http://www.xaml.net/

http://www-128.ibm.com/developerworks/library/specification/ws-secure/

http://www.oasis-open.org/specs/index.php#wssv1.0

http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=security

Google is your friend !!

Applicatie architectuur, toen

Applicatie architectuur, toen

Monolitisch

Een (set van) ‘exe’ / ‘library’ per applicatie

Een database (soms meer)

(vaak) 1 server (cluster)

Als 1 geheel ‘gereleased’ en onder 1 versiebeheer

Beheerd als 1 entiteit

Applicatie architectuur, toen

Opgebouwd volgens drie lagen model:

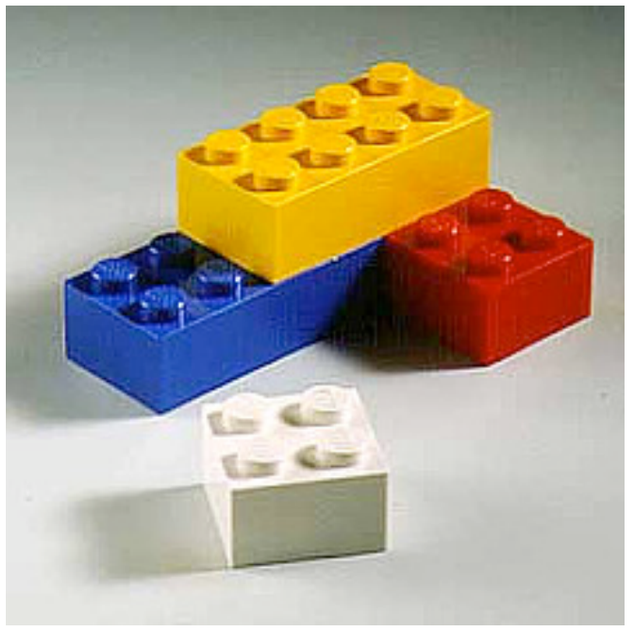
Database met gegevens

Business logic (web server)

Presentatie (web browser)

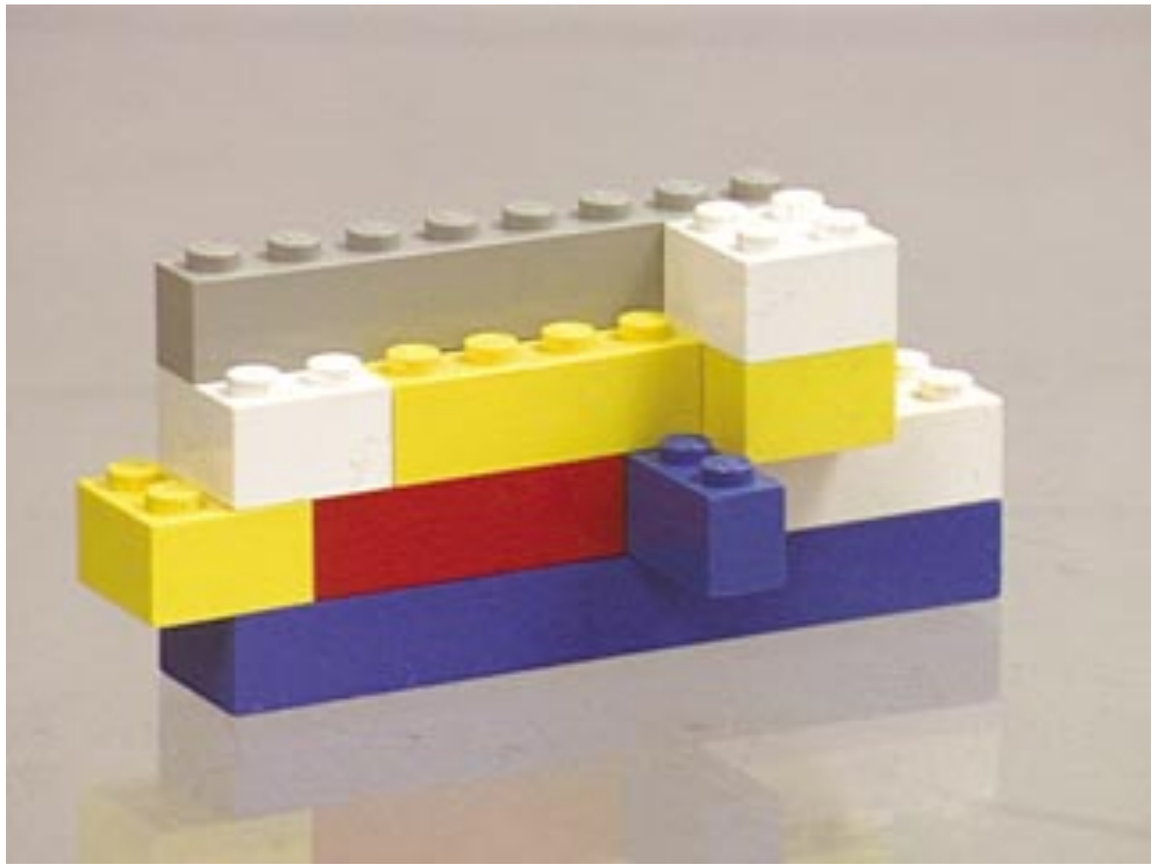
Wordt als 1 object door auditor gecontroleerd

Applicatie architectuur, nu



Bouwstenen

Applicatie



Applicatie architectuur, nu

- Service (data) gebaseerd
- Er is een set van (applicatie onafhankelijke) services (service bus)
- Service ontsluit data op applicatie onafhankelijke manier
- Service is het beheer object

Applicatie architectuur, nu

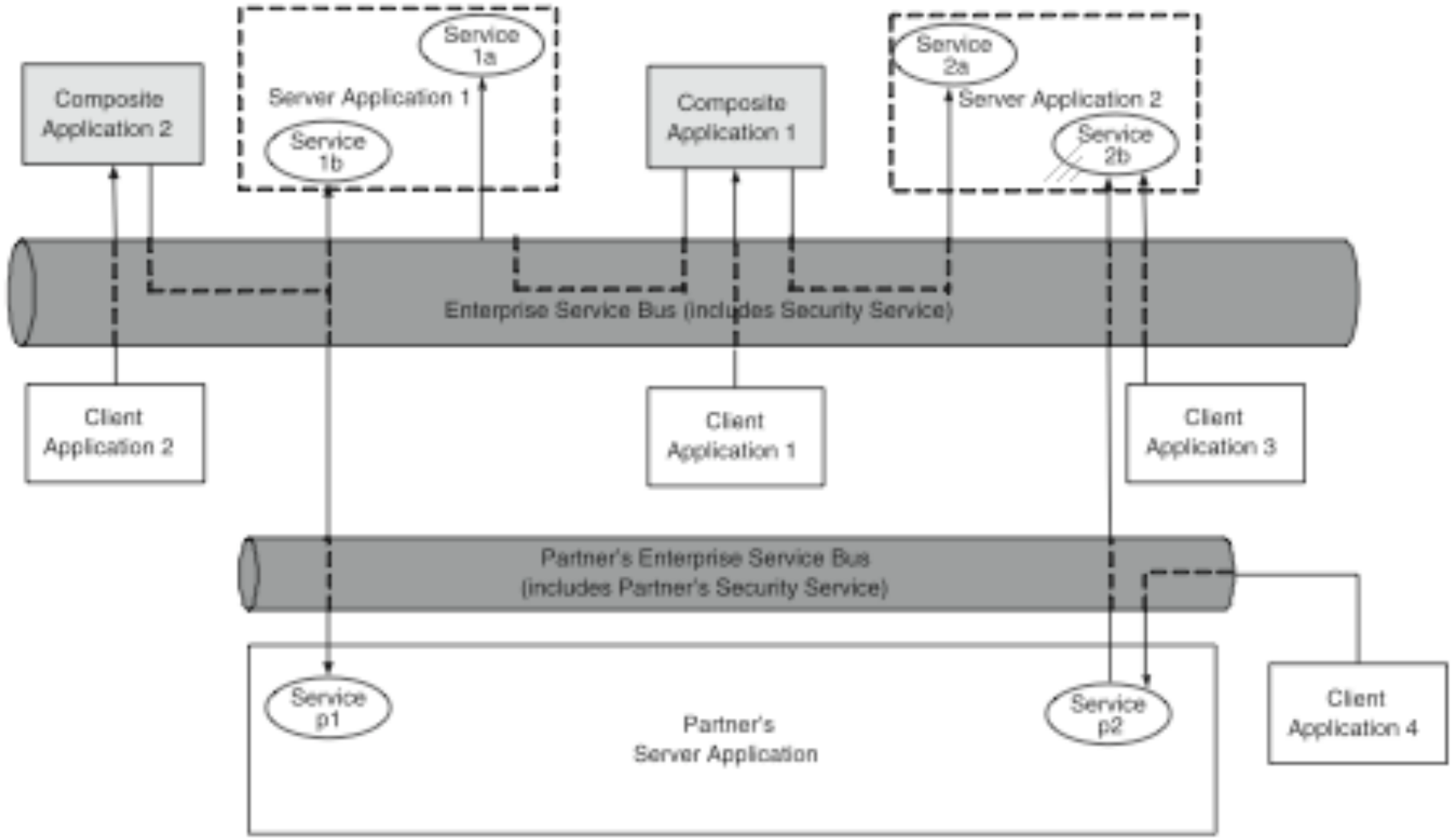
- De specifieke combinatie van services is de applicatie (bepaald door de business logica)
- Services kunnen zelf ook (deel van) business logica implementeren
- Services leven op de enterprise service bus
- Release en versie beheer per service

Applicatie architectuur, nu

‘Fuzzy’ lagen model:

Services combineren tot nieuwe services
Presentatie door middel van web server, XML portal of andere technologie
Wie is de data eigenaar (waar komt welke data vandaan ?)

Applicatie architectuur, nu



Applicatie architectuur, nu

Wat is nu object van onderzoek ?

De auditor....

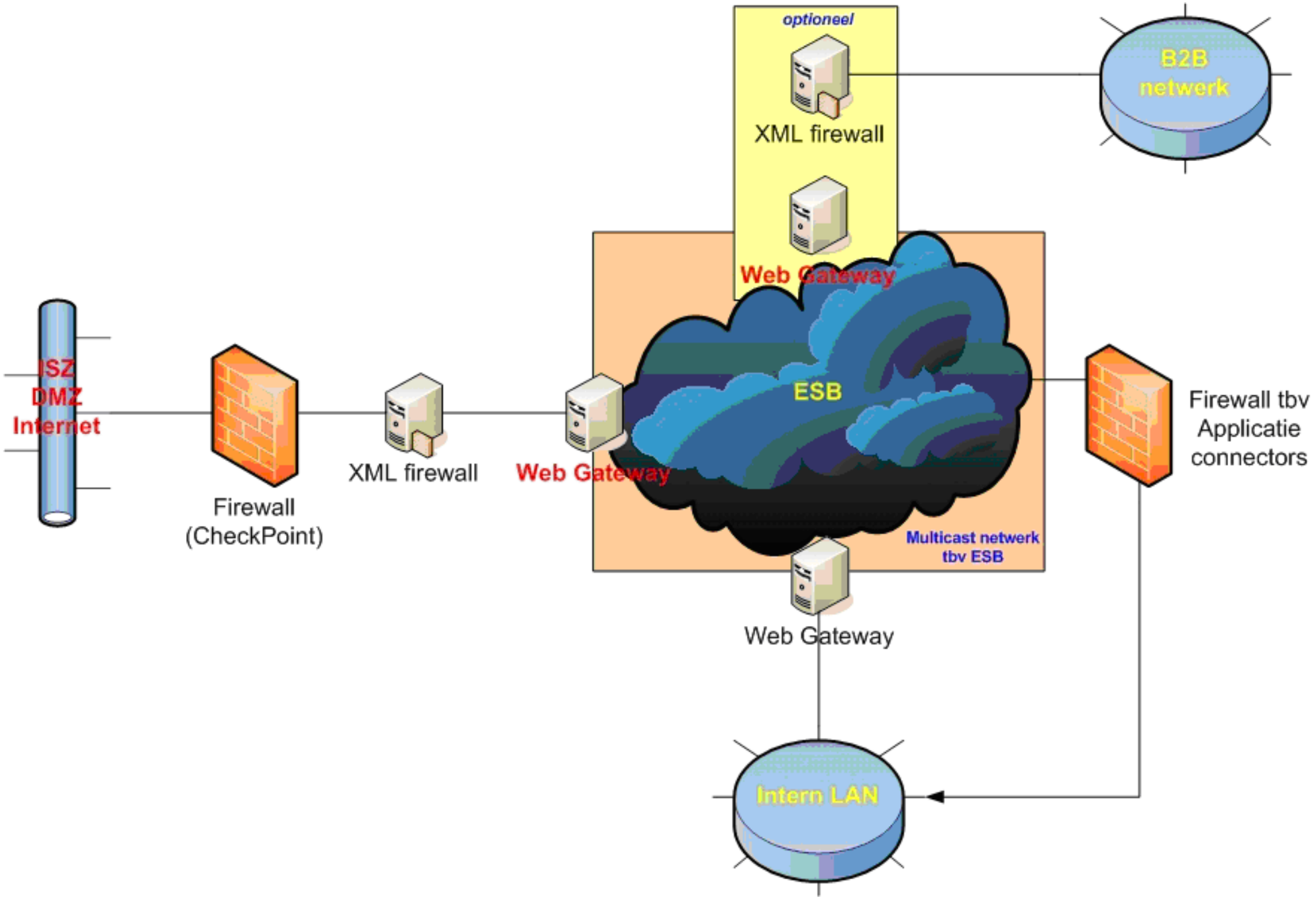
Praktijk case

- Ontsluiting van web services via portaal (WebSphere);
- Voor externe klanten (> 800.000), B2B partners en interne medewerkers;
- In de toekomst ook koppeling via XML gateway;
- Koppeling naar backoffice op basis van legacy (AS/400, Windows dmv JDBC);

Praktijk case security architectuur

- Verzoek om een security architectuur te maken
- Op basis van ISO27001 (17799-2005) een ‘bottom up’ risico analyse uitgevoerd;
- Is niet ideaal, maar binnen de tijd het meest pragmatisch (ivm volledigheid);
- Focus op basis aspecten, met extra aandacht voor logging (audit trail) en autorisatie.

Praktijkcase



Praktijk case security architectuur

Niet ideaal, maar werkbaar:

Gecontroleerde toegangspaden (piket-paaltjes), whitelisting in web gateway;

Authenticatie in toegangspad op user;

Authenticatie / autorisatie binnen ESB met systeem accounts plus applicatie accounts;

Audit trail door combinatie van logging op infrastructuur en applicatie niveau;

Praktijk case
Security architectuur

- Beschikbaarheid mbv multicast (product feature);
- Deel van security ingebed in ontwikkel-proces door middel van service templates;
- Frontend beveiliging door WebSphere of XML gateway (XSD validatie);

Ontwerp aspecten Vertrouwelijkheid

- Authenticatie op infrastructuur niveau (koppelingen) en binnen de service;
- Interne gebruiker authenticatie via portal en interne Active Directory;
- Externe gebruiker authenticatie in portal en externe provider (DigiD);

Ontwerp aspecten Vertrouwelijkheid

- Autorisatie primair binnen de service en applicatie;
- Authenticatie en autorisatie items zijn een onderdeel van de XML message layout (home grown SAML);
- Binnen de service vindt er nog data autorisatie plaats (niet iedereen die de service mag aanroepen mag alle data zien of bewerken);
- Mappen van gebruikers op rollen;

Audit aspecten Vertrouwelijkheid

- Audit / controle op koppelingen ten behoeve van authenticatie;
- Audit / controle op service ontwikkeling (automatische code review adhv template);
- Audit / controle op rollen en mapping (autorisaties);
- Audit / controle op (verborgen?) toegangspaden;
- Audit / controle op beheer;

Ontwerp aspecten Integriteit

- Versleutelde verbindingen (SSL) naar portal;
- Op ESB zelf geen aanvullende maatregelen (beperkt risico);
- Wel validatie controles in service zelf;

Audit aspecten
Integriteit

- Beperkte controle mogelijkheden vanwege beperkte implementatie (netwerk zelf biedt groot aantal waarborgen);
- Audit / controle op SSL koppelingen;

Ontwerp aspecten Beschikbaarheid

- Product zelf (Cordys) biedt beschikbaarheid door middel van dubbel uitgevoerde systemen;
- Ontsluiting naar portal via load balancers;
- Alles dubbel uitgevoerd en verdeeld over twee locaties;

Audit aspecten Beschikbaarheid

- Audit / controle op netwerk ontwerp;
- Audit / controle op uitwijk test (nog leuke netwerk uitdagingen :)

Ontwerp aspecten Controleerbaarheid

- Een betrouwbaar audit trail is moeilijk (allemaal losse services);
- Elke service dient zelf logging te verzorgen;
- Verzamelen op 1 centrale plek (denk aan tijd synchronisatie);

Audit aspecten
Controleerbaarheid

Audit / Controle op tijd synchronisatie;

Audit / Controle op volledigheid audit trail;

Audit / Controle op de code (geautomatiseerd);

Verbeterpunten ?

Ja zeker !

Op veel aspecten, wel gebonden aan de (on)mogelijkheden van het product.

Gebruik meer standaarden, in plaats van zelf bouwen (*security is a service*), dit is beter centraal te beheren en minder afhankelijk van ontwikkelproces.

Security service

SAML(2) (Security Assertion Markup Language):

"Security Assertion Markup Language (SAML) is an XML standard for exchanging authentication and authorization data between security domains, that is, between an identity provider and a service provider. SAML is a product of the OASIS Security Services Technical Committee."

Security service

WS Security

"This specification describes enhancements to SOAP messaging to provide message integrity and confidentiality.

The specified mechanisms can be used to accommodate a wide variety of security models and encryption technologies."

Security service

XKMS

Integratie van PKI in XML, ten behoeve van digitale handtekeningen (non repudiation) en message integriteit;

Applicatie ontwikkeling, toen

Applicatie ontwikkeling

Vroeger ‘simpel’:

1 applicatie == 1 release

Versie beheer voor de hele monoliet

O.T.A.P. scheiding

Omgeving per applicatie

Applicatie ontwikkeling, nu

Applicatie ontwikkeling

Nu niet meer zo ‘simpel’:

1 applicatie is meer dan 1 service, meer dan 1 release

Versie beheer per service

O.T.A.P. scheiding essentieel, maar soms moeilijk

Omgeving per service ?

Applicatie ontwikkeling

- Unit en integratie testen essentieel
- Versie beheer, zowel in ontwikkel als runtime omgeving
- Configuratie beheer
- Wijzigingen beheer
- Code reviews (geautomatiseerd)
- Templates voor services

Conclusies

- Nieuwe technologie en beveiliging is niet altijd een gelukkig huwelijk;
- Inpassen van een ESB in een bestaande omgeving vergt zorgvuldigheid;
- Audit objecten veranderen, benadering meer vanuit data dan applicatie;
- Standaarden zijn nog geen standaarden;

Vragen ?