Tracciabilità degli utenti in applicazioni multipiattaforma

Case Study assicurativo/bancario

Yann Bongiovanni
y.bongiovanni@integra-group.it
Roma, 4 ottobre 2006

INTEGRA

Retroscena

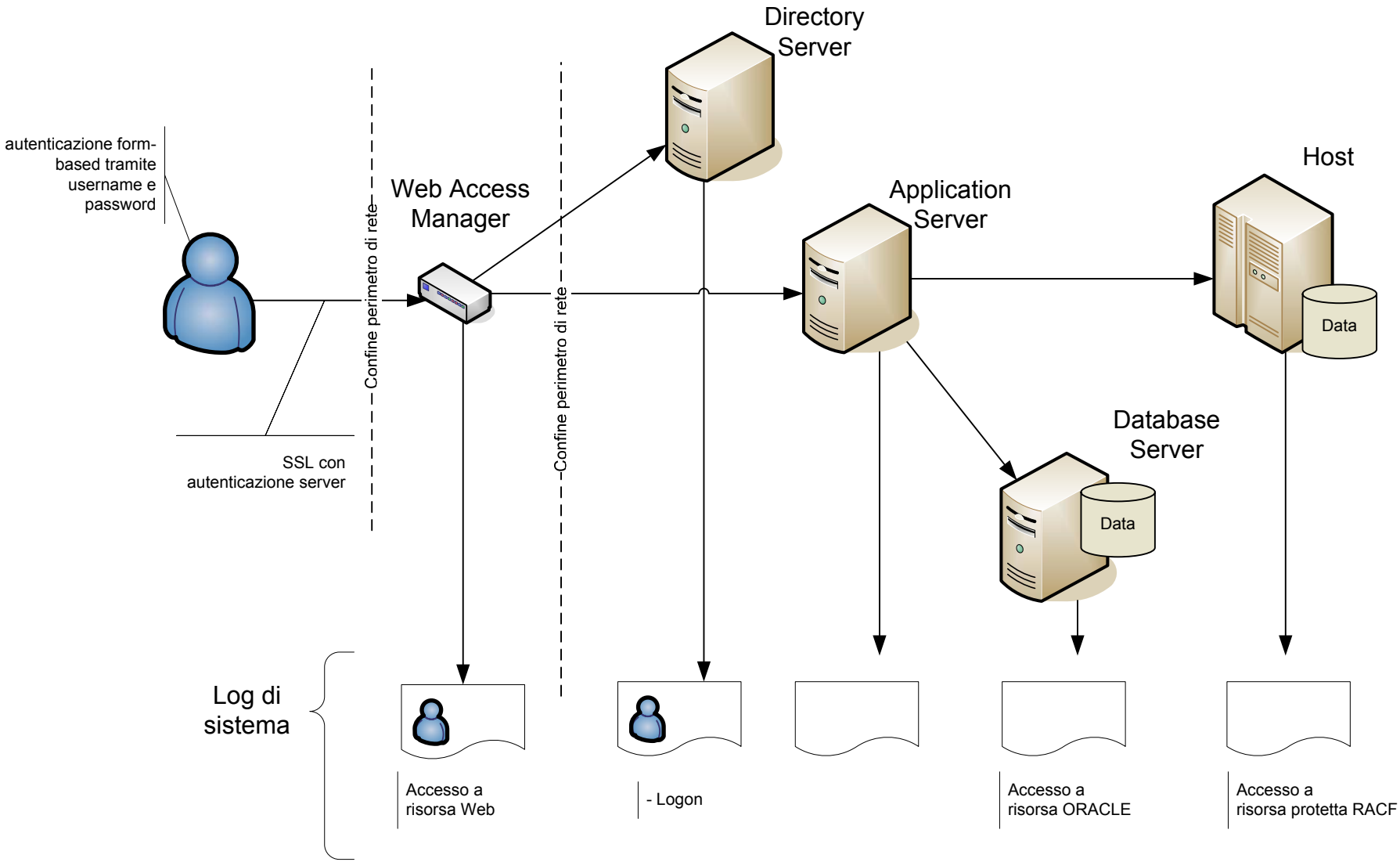
- Un’azienda multinazionale del settore assicurativo si trova a dover recepire nuove politiche di sicurezza derivanti dal Sarbanes-Oxley Act.
- Tramite un assessment della sicurezza determina le aree di vulnerabilità e non-conformità con le nuove politiche di sicurezza.
- L’assessment prende in considerazione gli aspetti di sicurezza di ca. 50 applicazioni critiche in dieci aree applicative.
- La maggior parte delle applicazioni è di tipo client-server, con l’80% della logica applicativa realizzata in programmi COBOL su mainframe.

Motivazioni

- Le nuove politiche di audit prevedono di
  - la necessità di tracciare accessi in scrittura
  - la necessità di tracciare accessi in lettura
  - la presenza di un log separato dall'applicazione

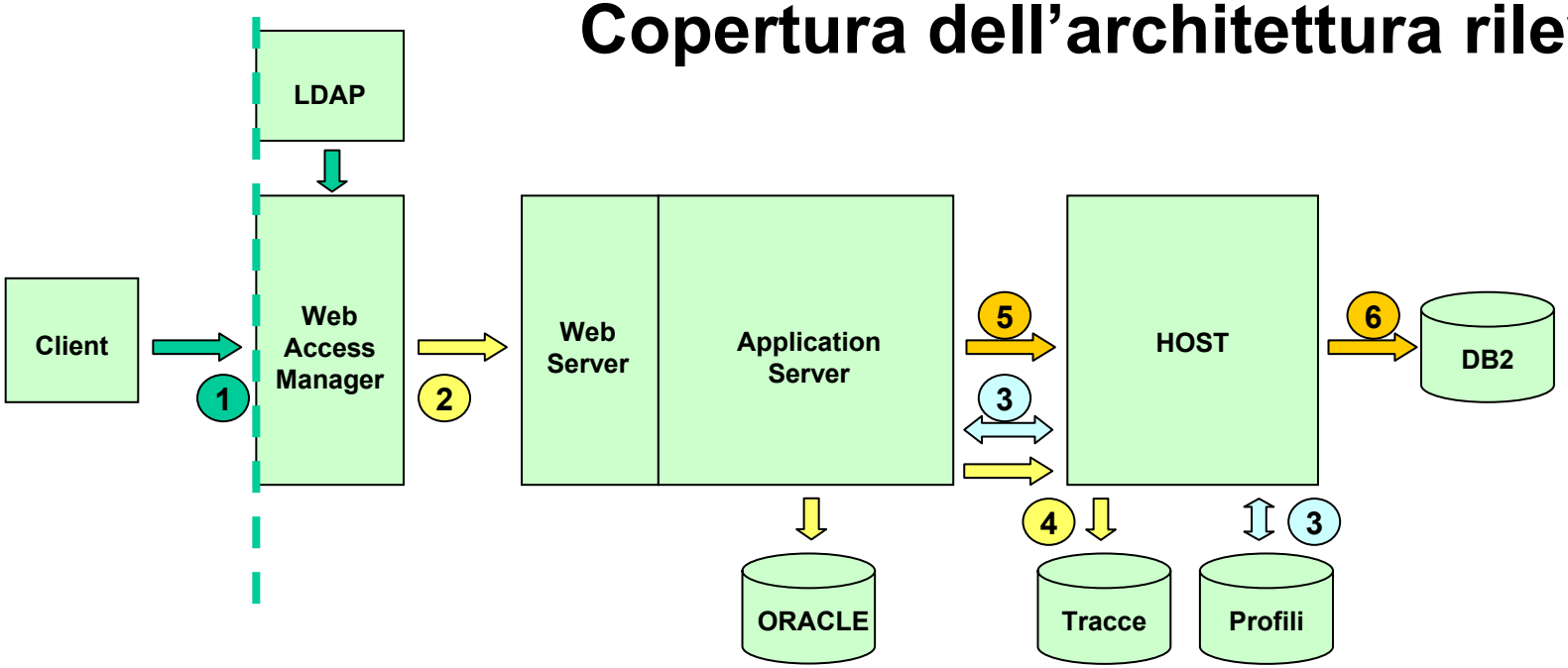
- Il sistema di generazione dei dati di audit presenta le seguenti caratteristiche:
  - Per alcuni tipi di operazioni, le applicazioni scrivono utente e marca temporale nelle tabelle applicative che "accompagnano" il dato.
  - Le anagrafiche contengono utente/data dell'ultima modifica.
  - I dati di audit sono sparsi nelle tabelle applicative.

Architettura rilevata



INTEGRA

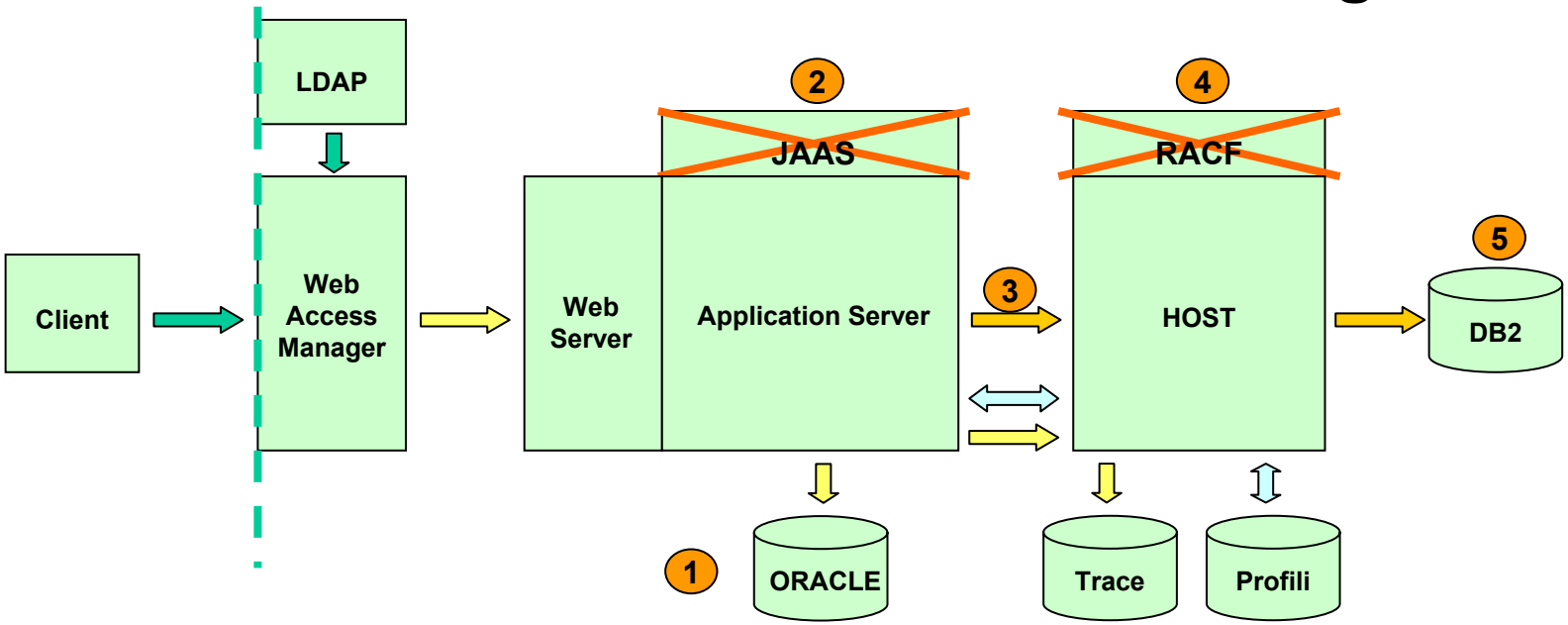
Copertura dell'architettura rilevata



1. Il Web Access Manager autentica l'utente interrogando LDAP e abilita solo i siti ammessi
2. Web Access Manager comunica all'architettura l'utente ed il codice del profilo applicativo
3. L'architettura legge il dettaglio del profilo da host e lo mette a disposizione dell'applicazione sull'application server
4. Una componente architetturale traccia gli eventi significativi (Login, Emissione polizze, Ristampe polizze), l'attivazione della componente è applicativa
5. L'utente viene trasmesso dall'architettura alla parte applicativa su host
6. L'applicazione traccia gli accessi a DB2 e le operazioni (se necessario)

INTEGRA

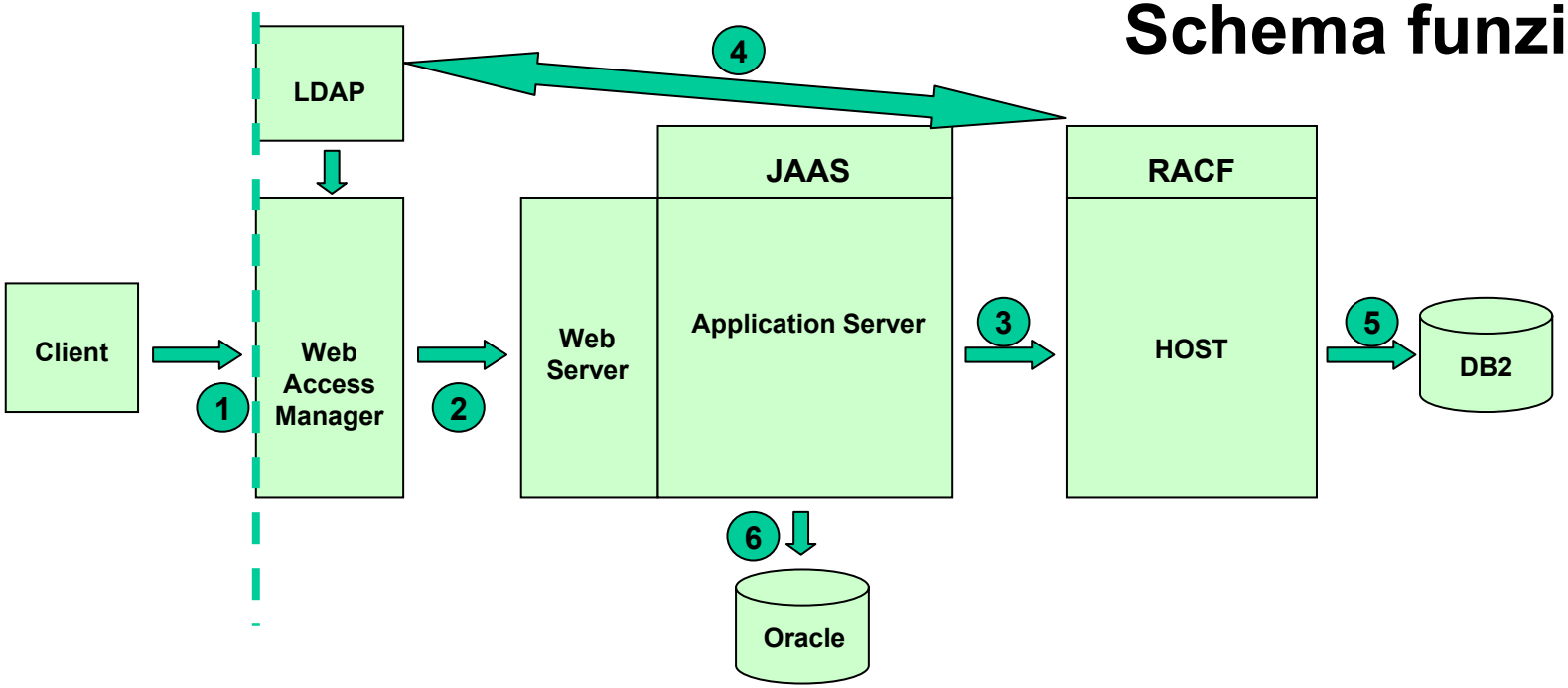
Aree di miglioramento



1. Il sistema ORACLE non è a conoscenza dell'utente connesso, non è attivo l'audit
2. L'utente all'interno dell'application server non è noto al sistema (JAAS) ma solo ad architettura ed applicazione
3. L'utente viene trasmesso ad host ma può essere utilizzato per il trace solo dalle applicazioni e l'impiego è facoltativo
4. Il sistema host non è a conoscenza dell'utente connesso, non è attivo il controllo di RACF sugli accessi
5. Il sistema di auditing sugli accessi a DB2 non è attivabile perché l'utente applicativo è sempre lo stesso per qualsiasi utente collegato

INTEGRA

Schema funzionale



1. WAM autentica l'utente interrogando LDAP e abilita solo i siti ammessi
2. WAM, con un protocollo di single sign-on, comunica l'utente al sistema di sicurezza Java (JAAS)
3. L'architettura comunica ad host in modo sicuro (trusted) l'utente connesso, tutte le operazioni su host sono tracciabili
4. Componenti architetturali determinano l'abbinamento tra l'utente RACF e l'utente LDAP
5. L'utente è noto al sistema host ed è possibile attivare le funzionalità di auditing di DB2 per tracciare l'accesso ai dati
6. L'architettura comunica l'utente ad Oracle ed è attivabile l'auditing del DB sugli accessi

INTEGRA

Propagazione dell'utenza verso l'AS

- Obiettivi
  - Eliminazione dell'accesso anonimo all'application server
  - Applicazione del controllo tramite l'autenticazione ed identificazione dell'utente finale
  - Utilizzo dei meccanismi standard Java per la gestione del soggetto utente
  - Integrazione con i meccanismi di tracing
- Soluzione adottata
  - Le risorse applicative (servlet o pagine JSP) sono accessibili unicamente previa identificazione.
  - Un profilo di protezione per le risorse da proteggere (ad es. tutte le URL invocabili) è associato ad un ruolo applicativo. Il ruolo applicativo è assegnato all'utente in fase di login.
  - L'identificazione può avvenire per propagazione (WAM) oppure per autenticazione diretta.
- Interventi:
  - Configurazione della user registry per l'interfacciamento al Directory
  - Configurazione del login al Java Authentication and Authorization Service (JAAS)
  - Attivazione dell'obbligo di identificazione ed autenticazione all'AS

Allineamento e mappatura degli identificativi utente Directory/RACF

- Obiettivi
  - Garantire la corrispondenza tra utenze gestite all'interno del Directory e di RACF
- Soluzione adottata
  - Se presente nella directory, l'utenza RACF deve essere associata al soggetto JAAS per consentire la propagazione verso i sistemi di back end.
  - La configurazione prevede l'inserimento di un modulo di login (JAAS) che effettua le seguenti operazioni:
    - recupera il nome utente che sta effettuando la login;
    - si collega al directory server per recuperare il nome utente RACF associato;
    - inserisce nel contesto di sicurezza il nome utente RACF
    - traccia l'avvenuta associazione (con i relativi nomi utente) ed i tentativi di associazione falliti (con la relativa causa).
- Interventi
  - Definizione e sviluppo di una procedura per la mappatura di utenze di agenzia con utenze RACF; integrazione nell'applicativo per la gestione delle utenze di agenzia.
  - Sviluppo di un "JAAS LoginModule" per l'aggiunta dell'identificativo RACF al contesto di sicurezza JAAS
  - Inserimento del LoginModule nella configurazione di login dell'Application Server

Propagazione dell'utenza da AS a RACF

- Obiettivi
  - Eseguire i servizi lato host con le autorizzazioni dell'utente finale
  - Generare una traccia delle associazioni Directory/RACF ad ogni occorrenza d'uso
  - Tracciare le attività dell'utente finale lato host
  - Garantire integrità tra il contesto di sicurezza lato AS e ed il contesto lato host
  - Ottenere il massimo livello di prestazione nonostante l'overhead causato dai meccanismi di protezione.
- Soluzione adottata
  - Creazione di un trusted path attraverso autenticazione bilaterale basata su certificati X.509.
  - La procedura di propagazione trasferisce il contesto di sicurezza dell'utente che invoca una funzione applicativa alla transazione eseguite lato host.
  - La componente che effettua le connessioni al CICS è autorizzata ad accedere al certificato digitale per l'autenticazione e rende disponibile le connessioni necessarie in un pool.
- Interventi
  - Modifica del componente client per realizzare il recupero e la trasmissione dell'utenza all'invocazione del servizio host
  - Modifica della transazione CWI ANALYZER per recuperare l'identificativo utente ed impersonificare l'utente nel CWI server
  - Definizione di una procedura per la generazione di certificati per l'autenticazione SSL tramite RACF
  - Attivazione dell'SSL lato CWI server con autenticazione bilaterale obbligatoria
  - Attivazione dell'autenticazione SSL bilaterale nel client
  - Realizzazione del connection pooling nel client

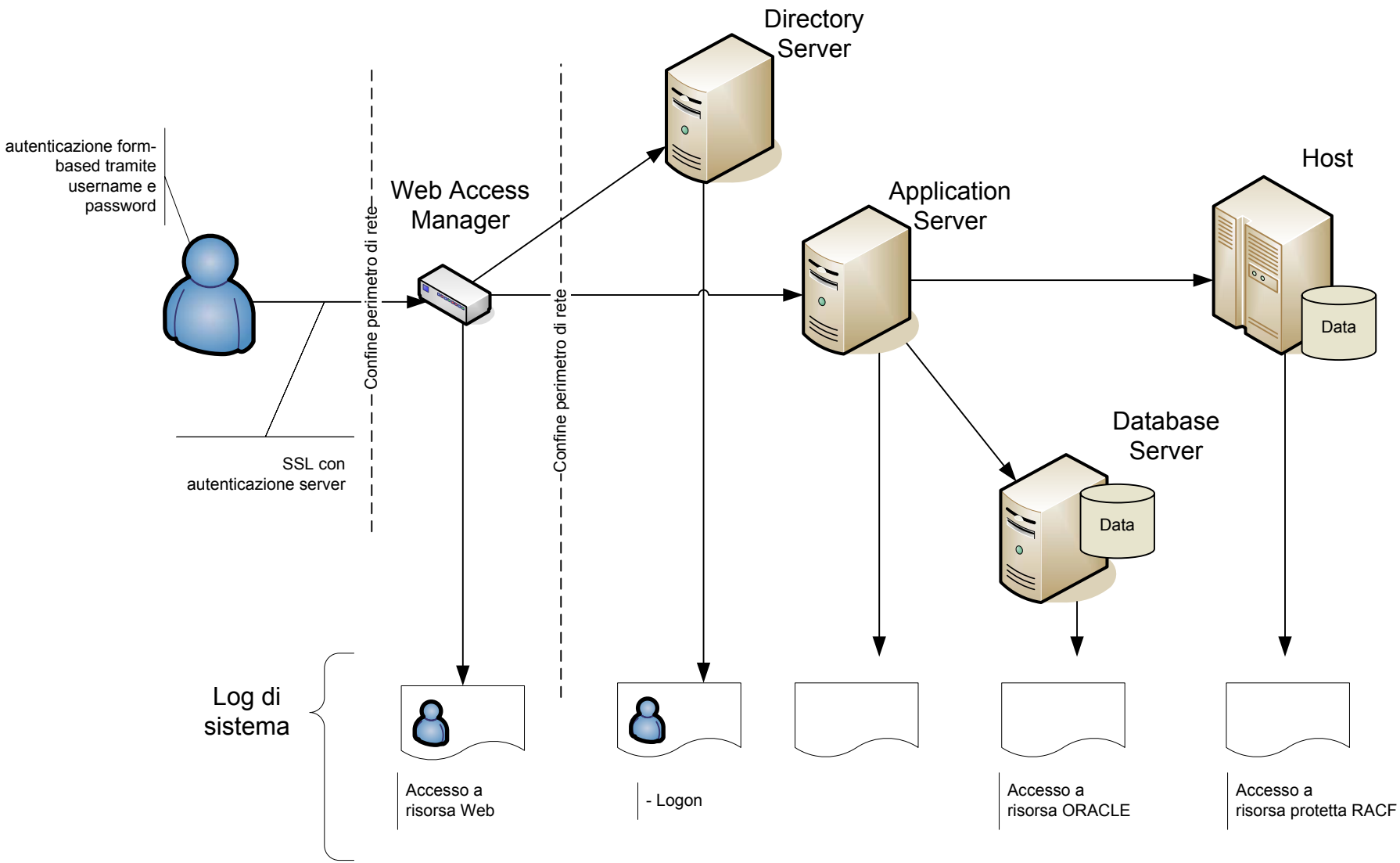
Propagazione dell'utenza da WebSphere a ORACLE

- Obiettivi
  - Trasferire al database l'identificativo utente in modo utilizzabile dal sottosistema di audit
  - Tracciare le attività dell'utente finale lato ORACLE
  - Proteggere il canale garantendo l'integrità tra il contesto di sicurezza lato AS e ed il contesto lato ORACLE
  - Ottenere il massimo livello di prestazione nonostante l'overhead causato dai meccanismi di protezione.
- Soluzione adottata
  - Realizzazione di un trusted path attraverso il meccanismo di autenticazione basato
    - su certificati digitali (client e server) per l'autenticazione del server e del client,
    - sulla combinazione username/password per l'autenticazione dell'utente.
  - La propagazione dell'utenza dall'application server verso il DB si realizza valorizzando la sessione CLIENT_IDENTIFIER prima dell'utilizzo della connessione (dopo che essa è stata recuperata dal connection pool). Al termine dell'utilizzo, prima del rilascio della connessione al connection pool, l'attributo di connessione viene annullato
- Interventi
  - Recupero e trasmissione dell'utenza tramite CLIENT_IDENTIFIER
  - Configurazione dell'SSL nella connessione verso ORACLE

Protezione delle risorse critiche dell'architettura

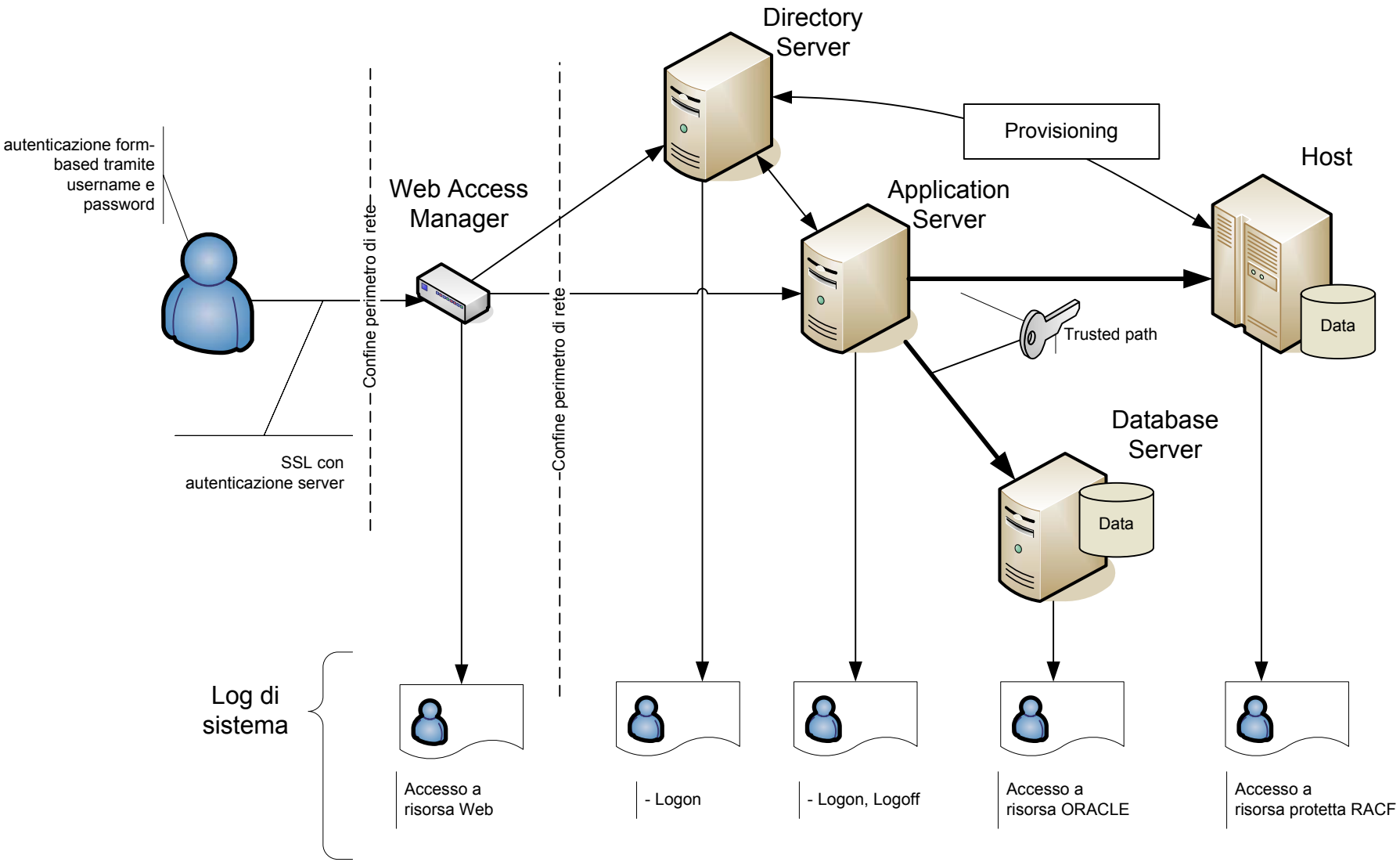
- Obiettivi
  - Impedire che le applicazioni che fanno uso dell'architettura possano eludere io manipolare meccanismi di propagazione dell'utenza
  - Vincolare l'applicazione all'uso delle componenti di architettura per la comunicazione con i sistemi di back-end
- Soluzione adottata
  - Utilizzo del sistema di controllo accesso "Java 2 Security" per la protezione delle risorse critiche
- Interventi
  - Sviluppo di un tool per l'analisi delle autorizzazioni richieste dall'applicazione
  - Analisi delle autorizzazioni in contesto Tomcat e WebSphere
  - Realizzazione ed implementazione della Java Policy per la protezione delle risorse dell'architettura

Situazione precedente



INTEGRA

Situazione al termine dell'intervento



INTEGRA

Considerazioni e futuri sviluppi

- Considerazioni
  - La nuova architettura permette di impostare la configurazione di audit partendo dall'utente e/o dal dato (ignorando l'applicazione che vi accede).
  - Il progetto si è svolto in un arco temporale di 6 mesi senza l'impegno di risorse a tempo pieno.
  - La presenza di una architettura applicativa uniforme con componenti specializzate per l'accesso a DB e host ha permesso la realizzazione delle modifiche senza dover modificare le applicazioni.
  - Nella nuova architettura di sicurezza l'applicazione non è autorizzata ad intervenire sulla configurazione di identificazione e audit.
  - La nuova architettura permette di realizzare la segregazione dei ruoli tra chi sviluppa l'applicazione, chi ne gestisce l'esercizio e chi ne configura la sicurezza.
- Futuri sviluppi
  - Per migliorare la granularità dell'audit si sta procedendo per trasferire la gestione della sicurezza DB2 a RACF.
  - Per migliorare il controllo accessi nelle applicazioni di front end si sta valutando l'utilizzo di un framework di controllo accessi open source.
  - E' in fase di studio un sistema di gestione centralizzata e correlazione dei dati di audit ora presenti.