SAMLAD INLOGGNING

- hur Samlad Inloggning används inom företag

Niklas Björk, 33077
Kandidatavhandling i datateknik
Åbo Akademi
Institutionen för Informationsteknologi

Referat

Samlad Inloggning (Single Sign-On, SSO) är benämningen för tekniken som eliminerar kravet på att användarna skall ha flera lösenord för att komma åt de olika dataresurserna inom ett eller flera nätverk. Detta innebär att data hjälpcentralen kommer att få mindre arbeten angående lösenords återställning och data administrationen behöver inte investera lika mycket tid på att upprätthålla användar databaser hos de olika data resurserna. I denna avhandling så tas upp olika tekniker som möjliggör Samlad Inloggning och dessa tekniker förklaras mera ingående. Dessutom tas olika Samlad Inloggning arkitekturer upp och deras för- och nackdelar jämförs.

Nyckelord: Samlad Inloggning **(**Single Sign-On, SSO), Web samlad inloggning, Legacy/Enterprise samlad inloggning, Gemensam samlad inloggning, Autentisering, Auktorisering

Innehållsförteckning

1. Inledning

Samlad Inloggning blev uppfunnen på grund av diverse orsaker. En orsak var att man ville minska på antalet lösenord som en användare måste komma ihåg för att ha tillgång till alla data resurser i ett nätverk. En användare kan ha upp till ett dussin olika lösenord att komma ihåg och detta ledde till att man började skriva ner dessa lösenord på ”komihåg” lappar. Dessa ”komihåg” lappar sparad man sedan ”lätt åtkomligt” vid sin arbetsstation vilket utgjorde en allvarlig säkerhets risk. En annan orsak till att Samlad Inloggning blev uppfunnen var att man ville lätta på IT-stödets arbetsbörda genom att minska på antalet incidenter som de får för att återställa glömda lösenord. Istället kan de koncentrera sig för verkliga IT problem.

Samlad Inloggnings lösningar är indelade i två huvud grupper, nämligen de som hanterar bara en uppsättning av autentiseringsuppgifter och de som hanterar Flera uppsättningar av autentiseringsuppgifter [3]. Dessa lösningar är baserade på redan existerande teknik.

I detta arbete kommer jag att presentera olika SSO arkitekturer och jämföra dessas med varandra genom att ta upp fördelar och nackdelar hos dem. Jag tar också upp Identitets Synkroniserings lösnings alternativet även om den inte är en sann Samlad Inloggnings lösning.

2. Vad är Samlad inloggning?

Samlad Inloggning är tekniken som ger användaren möjligheten till ett engångs autentiserings process och sedan ha tillgång till andra skyddade resurser utan att behöva autentisera sig igen. Open Group definierar Samlad Inloggning som den mekanism där en ända handling av användarautentisering och auktorisering kan tillåta användaren att komma åt alla datorer och system var användaren har åtkomstbehörighet, utan att behöva ange flera lösenord [3]. Det bör klargöras att ordet ”användare” skall tolkas i en bred bemärkelse: det täcker alla säkerhetsobjekt som har tillgång till de resurser som är under auktoriserings auktoritets kontroll. Open Group är ett globalt konsortium vars uppgift är att möjliggöra uppnåendet av verksamhetsmål genom IT-standarder.

Ett typiskt företags användare måste autentisera sig flera gånger för att kunna få tillgång till diverse applikationer som han/hon behöver för att kunna utföra sitt arbete. Från användarens synvinkel så är dessa upprepade autentiseringar och kravet på att komma ihåg flera olika lösenord de ledande orsaker till att de upplever missnöje av att använda företagets datatjänster. Enligt IT-administrationen så ökar glömda lösenords incidenter markant på administrations kostander.

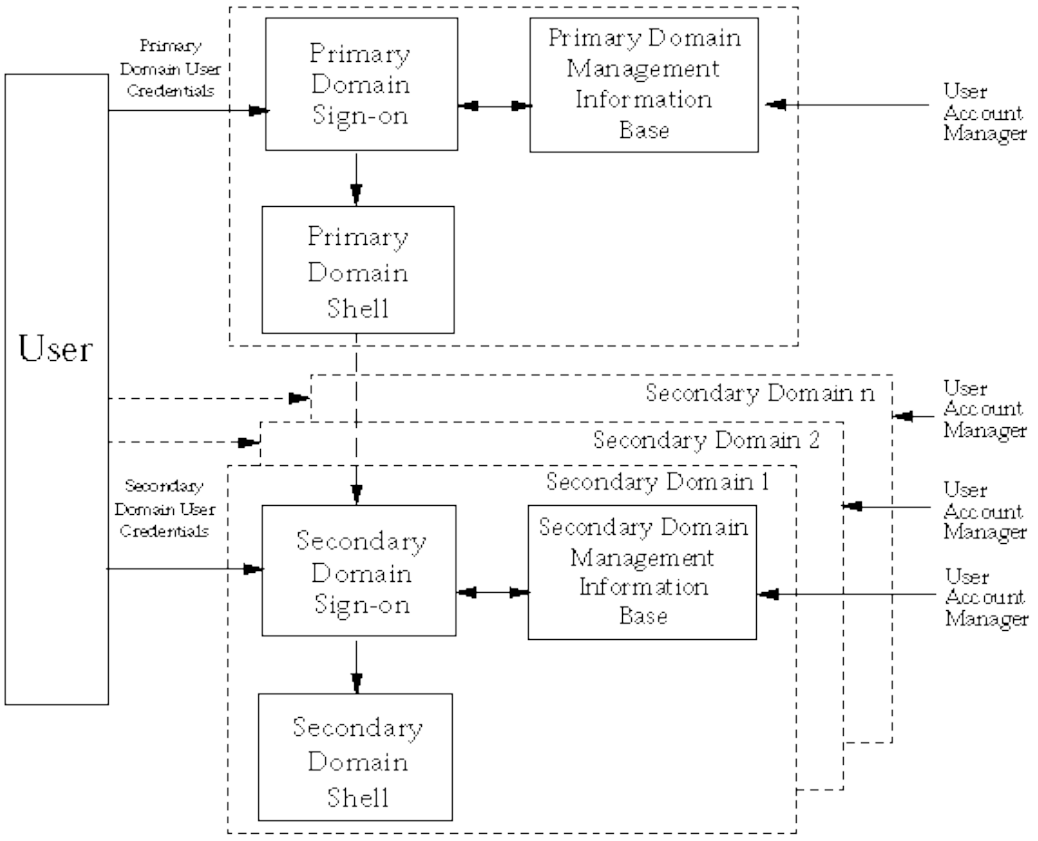
På grund av de till synes oöverkomliga problemen som flera autentiserings identiteter utgör, så har konceptet Samlad Inloggning blivit den ” Heliga Graalen” för identitets hanterings projekt[6, 8].

2.1 Hur det fungerar

Figur 1 illustrerar ett nätverk som man inte har implementerat en Samlad Inloggnings lösning till och i detta fall så hamnar användaren att upprepade gånger autentisera sig för att nå alla de data resurser som han/hon behöver. Då användaren autentiserar sig för första gången så sker det till en så kallad primär

domän, alltså till den domän som är förvald för användaren. För att autentisera sig till denna domän så måste användarnamn och lösenord ges till den primära domänen som sedan verifierar i sin databas att dessa autentiseringsvärden stämmer för den ifrågavarande användaren. Om autentiseringsvärden var korrekta så får användaren tillgång till alla de dataresurserna som finns till förfogande hos denna domän. Men om användaren är ute efter ett specifikt program eller tjänst som inte finns på denna domän utan på en annan, så hamnar han/hon att ta kontakt med någon annan domän. För att få tillgång till denna dataresurs på den andra domänen så måste användaren autentisera sig igen. Följande domäner som användaren skall autentisera sig till kallas för sekundära domäner.

Figur 1 illustrerar också hur komplext det är att upprätthålla användarkonton för varje enskild domän. Varje domän har sin egen databas som innehåller användaruppgifter och rättigheter på alla de användare som skall ha tillgång till domänen. Om en användare glömt sin lösenord och vill att den skall nollställas så måste IT-stödet manuellt ta kontakt med varje domän och nollställa den specifika användarens lösenord. Samma genomgång gäller om användaren slutar jobba hos företaget, men istället raderas kontot från databasen.



Figur 1[7] Användaren hamnar att autentiserar sig till olika domän

Figur 2 illusterar ett nätverk var man har implementerat en samlad inloggnings lösning. I detta fall måste användaren autentisera sig endast en gång och detta sker via primärdomänen. Efter en lyckad autentisering har användaren full tillgång till alla domän i nätverket och behöver inte autentisera sig flera gånger. I samma figur ser vi att alla sekundära domäner litar på den primära domänen till vilken användaren autentiserat sig. Primära domänen tar upp användarens autentiserings uppgifter och kan använda dessa för autentisering till de sekundära domänerna. Sekundär autentisering kan ske i olika former.

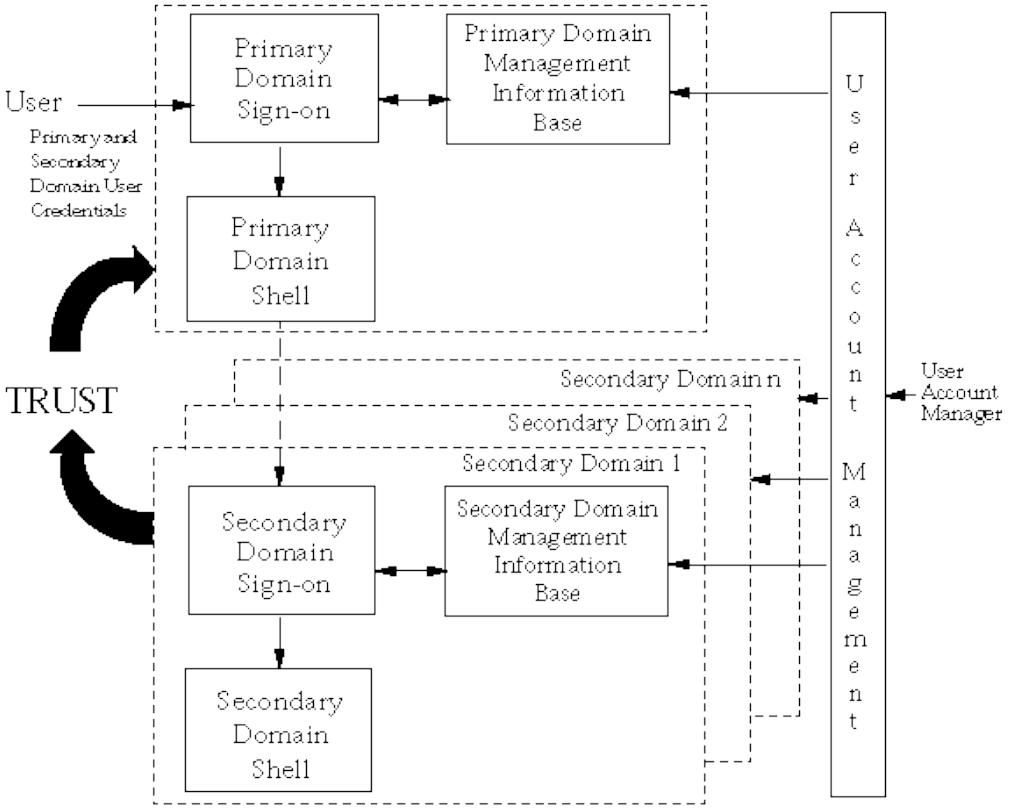
Det kan ske direkt, var informationen som användaren gett överförs till sekundära domänerna i form av sekundär autentisering [7].

Det kan också ske indirekt, var informationen som användaren gett används till att hämta andra användaridentifierings och autentiserings uppgifter vilka är sparade i samlade inloggningsanvändardatabasen. Den hämtade informationen används sedan till sekundär domän autentisering. [7]

Det kan ske omedelbart, för att skapa en session med de sekundära domänerna som en del av den ursprungliga sessions upprättning. Detta innebär att program klienter anropas automatiskt och att kommunikationen fastställs vid samma tidpunkt som den primära autentiseringen [7].

Det kan också ske stegvist dvs. att man sparar i minnet eller cashar användaridentifiering och autentiseringsvärden som används då en åtkomst begäran för en sekundär domän gjorts av slut användaren [7].

En samlad inloggnings lösning som denna kommer också att underlätta avsevärt på IT stödets arbete då de har bara ett enda databas att administrera. Denna databas synkroniseras automatiskt till företagets alla andra domän.



Figur2 [7] Användaren autentiserar sig till primära domänen och har efter det tillgång till alla andra domän

2.2 Fördelar med SSO

The Network Applications Consortium gjorde en undersökning på stora företag och resultaten visade att användarna utför inloggnings aktiviteter i medeltal 44 timmar per år för att få tillgång till fyra program. Samma undersökning mätte innehållet på samtal som IT-supporten fick och det visade sig att 70 % av alla samtal var lösenords nollställnings relaterade samtal [3]. Så med andra ord så kommer en samlad inloggnings lösning att öka på användarnas produktivitet och spara på IT-administrations kostnader.

En Samlad Inloggnings lösning kommer också att ge möjligheten för företage att stärka sin datasäkerhet med att implementera en starkare lösenords policy. Även om förtag inte vill ta detta steg så kan användarna ändå välja en mera komplex lösenord eftersom de hamnar att bara komma ihåg ett ända lösenord.

Användarna behöver bara autentisera sig en gång per session för att få tillgång till alla domäner som han/hon behöver, dessutom så ökar produktiviteten på grund av denna lösning.

2.3 Nackdelar med samlad inloggning

Ett ofta mött argument mot Samlad Inloggning är att Samlad Inloggnings autentiseringsuppgifter ligger bakom ett så kallad huvud nyckel Alltså om någon får tag på någon persons Samlad Inloggnings autentiseringsdata, får denna tillgång till alla system som är säkrade av Samlad Inloggnings system [3]. Ett annat scenario är att om en användare lämnar sin arbetsstation utan att logga ut sig från sin session så kan en obehörig person få tillgång till alla resurser den inloggade personen har tillgång till och möjligen orsaka obehörig intrång och skada. Dessa risker kunde reduceras med datasäkerhets åtgärder som t.ex. användarens automatiska utloggning efter en viss tid samt att man väljer samlade inloggnings autentiseringsuppgifter vilka inte är kunskapsbaserade (lösenord är ett typiskt exempel) utan biometrik baserade uppgifter (till exempel fingeravtryck) eller innehavs baserad (ett kryptografiskt token eller ett smart kort). Autentisering med hjälp av flera användaruppgifter (alltså en kombination av kunskaps baserade uppgifter biometriska uppgifter och innehavs baserade) kommer ytterligare reducera risken [3].

3. Teknik som möjliggör samlad inloggning

3.1 Webb baserad samlad inloggning

Webb baserad Samlad Inloggning (Web SSO) används då man har web resurser till förfogande. En användare som försöker ta kontakt med någon web resurs via sin webbläsare kommer att bli dirigerade till en central autentiseringsserver var han/hon kommer att ge sina inloggningsuppgifter. Detta sker därför att de inte är autentiserade till domänen ifråga. Efter att de blivit autentiserade, kommer de att

få en cookie vilket indikerar att de blivit autentiserade till domän. Efter detta så blir användaren dirigerade tillbaka till web resursen var användaren sedan presenterar cookien, som den fick från autentiseringsservern. Webb resursen kommer att kommunisera med den centrala autentiseringsservern för att ta reda på om cookien är i kraft, om sessionen är ännu aktiv och vilka rättigheter användaren har. Efter att användaren blivit autentiserad och auktoriserad, kommer han/hon att ha fullständig tillgång till alla Webb resurser som han/hon har tillåtelse till [8, 9].

3.2 Legacy / Enterprise Samlad Inloggning

Legacy / Enterprise Samlad Inloggning är mycket likt Webb baserad Samlad Inloggning för att båda är designade att hantera flera inloggningar till applikationer efter en autentiserings händelse [3, 4]. Legacy / Enterprise Samlad Inloggning funktionalitet inkluderar web resurser och dessutom äldre applikationer inom ett företags interna nätverk.

3.3 Gemensam inloggning (Federated SSO)

Ett företag kan ha flera kunder eller affärs partners som behöver ha tillgång till en eller flera av företagets dataresurser. Låt oss säga att vi har en kund som vill logga in till företaget AB:s system. När kunden klickar på en länk till företag AB så gör kundens Samlad Inloggnings system en säkerhets försäkran och vidarebefordra denna. Företag AB:s Samlad Inloggnings system kommer att ta emot denna försäkran, granska den och om den godkänns så kommer systemet att bevilja kunden tillgång till dataresurser utan att han/hon behöver logga in [8, 10].

För att få ett mera praktiskt exempel på Gemensam inloggning så kan vi undersöka Shibboleth systemet. Shibboleth är en samordnad identitetshanterings system. Den utgör en pålitlig metod för autentisering och auktorisering till en rad av olika leverantörer som alla är medlemmar i den gemensamma samordningen.

Det kan nämnas att Shibboleth systemet används också av Åbo Akademi. Denna följande förklaring finns bättre förklarat i [5]. Systemet består av fyra

huvudsakliga enheter nämligen Identity Provider (IdP), Service Provider (SP), Discovery Service (DS) och användaren. IdP:n och SP:n kommunicerar med varandra i form av metadata filer, vanligen som XML-filer (Extensible Markup Language) för att mera noggrant kunna identifiera leverantören och vilka service den erbjuder. För att förstå hur Shibboleth login fungerar så förenklar vi detta med ett ”begäran för en resurs” process. Användaren begär efter en resurs via en Service Provider (SP) och den kan ha två möjliga åtkomsträttigheter, d.v.s. skyddad eller inte skyddad. I det första fallet så är användaren dirigerad till Discovery Service DS var användaren sedan med hjälp av en grafiskt användargränssnittväljer sin IdP. Om användar autentiseringen lyckas så kommer hem domän IdP:n att konstruera en session och ett handtag [5]. I största delen av fallen så kommunicerar handtaget med användarens webbläsare och överlämnar handtaget till SP:n som i sin tur använder handtaget för att begära användar attribut från dess hem domän IdP. De begärda attributen från ett källsystem ges sedan till SP:n efter att den passerat ett rad huvudsteg. Det är viktigt att lägga märke till att IdP:n inte sparar några attributer av användaren. Den litar på att externa datalager tillhandhåller användare informationen för utgivning. Det finns möjligheten att data källan innehåller ett antal flera attribut om användaren. Lanseringen av attributen till SP:n beror på SP:s politik vilket anger vilka attribut SP:n behöver om användaren för att framställa tillgångs politik. IdP:n använder sig av attriubute-filter.xml filen för att definiera vilka attributer som kommer att lanseras till vilka SP:n. Genom att känna till basen för begärda och mottagna attributer så bestämmer SP:n att antingen ger eller nekar tillgång till det skyddade resursen/tjänsten. Ytterligare så är det viktigt lägga märke till att enligt DS kod så måste vi ha flera än en IdP för att installera och konfigurera DS:n för IdP:n. Detta är logiskt för att om vi har bara en IdP (vilket är fallet i största delen av fallen) då finns det ingen behov för en DS. En DS behövs då det finns då det finns flera än en IdP och att vi vill välja själv vilken vi vill använda [5].

4. Olika Arkitektur

Samlad Inloggnings arkitekturer är delad i två huvud grupper, de som hanterar bara en uppsättning av autentiserings uppgifter och de som hanterar flera uppsättningar av autentiserings uppgifter [15].

4.1 En uppsättning av autentiseringsuppgifter

De enklaste komplexa Samlade Inloggnings arkitekturer är de system som hanterar en uppsättning av autentiseringsuppgifter. Det finns två olika Samlad Inloggnings arkitekturer som hanterar en uppsättning av autentiseringsuppgifter, nämligen markör baserad Samlad Inloggning och Public key infrastructure (PKI) baserad Samlad Inloggning.

Båda av dessa lösningar förser Samlad inloggning i en homogen omgivning, alltså en omgivning var alla enheter, applikationer och tjänster deltar i samma Samlad Inloggnings omgivning genom att använda samma konto namngivnings format och autentiserings protokoll [11].

4.1.1 Markör baserad Samlad Inloggning

Figur 5 illustrerar hur en Markör baserad Samlad Inloggnings lösning fungerar. Då användaren autentiserat sig till den primära autentiseringsauktoriteten så får användaren en mjukvara markör som kommer att sparas i användarens maskinminne. Denna markör kommer att användas för att bevisa användarens identitet för de sekundära autentiseringsauktoriteterna. För att validera markören så används kryptering i form av privata nycklar (symmetrisk kryptografi) mellan primära och sekundära autentiserings auktoriteten. Denna kryptografi representerar förtroende mellan primära och sekundära autentiserings domänerna.

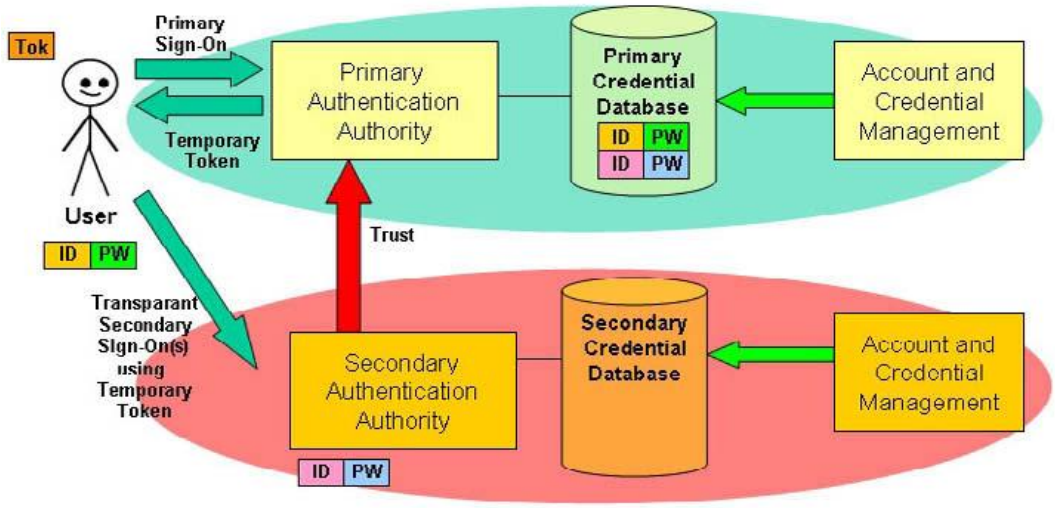


Fig. 3: [11] Markör baserad Samlad Inloggning

Det kanske mest kända markör baserade Samlad Inloggnings lösningen är den som utvecklades av av Massachusetts Institution of Technology (MIT), Kerberos. Den blev döpt efter den tre hövdade hunden som vaktar porten in till Hades i den grekiska mytologin. Kerberos används då en användare försöker få kontakt med en nätverks tjänst och tjänsten kräver autentisering. Efter att användaren autentiserat sig till Kerberos så får användaren en biljett från den och denna biljet innehåller information som länkar den till användaren. Användaren presenterar biljetten till nätverken för att få tillgång till tjänsten och näteverket analyserar biljetten för att verifiera användarens identitet och om användar verifieringen lyckas så får användaren tillgång till tjänsten [1, 2].

4.1.2 PKI baserad Samlad Inloggning

Figur 6 representerar hur en PKI baserad Samlad Inloggnings arkitektur fungerar. Först hamnar användaren att registrera sig till autentiserings auktoriteten och det kan ske på två olika sätt. För PKI arkiteturen så kallas autentiserings auktoritetet för en certifikatutfärdare (certificate authority, CA) och användaren kan registrera sig dit eller sen kan användaren registerar sig till en av certifikatutfärdarens flera registraturfunktioner (Registration Authority, RA). Under registrerings processen sker det flera olika saker: användaren identifierar sig genom att använda en uppsättning av identifieringsvärden; en del av klient mjukvaran genererar en asymmetrisk nyckel par; och den öppna nyckeln kommer att presenteras åt certifikatutfärdaren eller registraturfunktionen för certifiering. Vid mottagning av

användarens autentiseringsvärden och öppna nyckeln kommer certifikatutfärdaren eller registraturfunktionen att verifiera användarens autentiseringsuppgifter. Om autentiseringsuppgifterna var korrekta kommer en öppen nyckel certifikat att generaras och detta certifikat kommer att skickas tillbaka till användaren var den kommer att sparas i användarens maskinminne. Dessa uppgifter används sedan för att producera en mjukvara markör som används för att bevisa användarens identitet till de sekundära autentiserings auktoriteten. Dessa mjukvara markörerna liknar de som produceras i Markör baserad Samlad Inloggning.

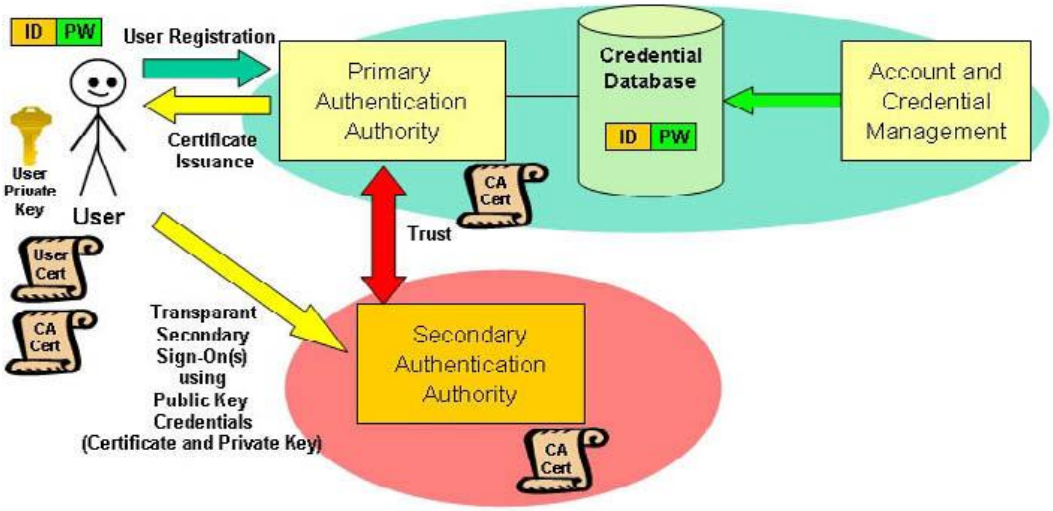


Fig. 4: [11] PKI baserad Samlad Inloggning

Den största skillnaden mellan markör baserad och PKI baserad Samlad Inloggning är att PKI lösningen använder asymmetrisk kryptografi för att verifiera användarens markör [2, 3].

4.2 Flera uppsättningar av autentiseringsuppgifter

Det finns tre stycken arkitekturer som kan hantera flera uppsättningar av autentiserings uppgifter. Dessa är: Identitetsuppgifts synkronisering, Secure Client-side Credential Caching och Secure Server side Credential Cashing, varav Identitetsuppgifts synkroniserings lösning räknas inte till ett verkligt Samlad Inloggnings lösning. Dessa arkitekturer kan förverkliga Samlad Inloggning i en mer heterogent omgivning [3].

4.2.1 Secure Client-side Credential Caching Samlad Inloggning

Figur 3 illusterar hur Secure Client-side Credential Caching Samlad Inloggning fungerar. I denna lösning autentiseras användaren genom att ange ett par av de primära autentiseringsvärdena. Efter autentiseringen får man tillgång till användarens identitets cache. Denna identitets cache finns sparad på användarens lokala maskins hårdskiva och innehåller användarens alla identitetsuppgifter som krävs för att man skall få tillgång till de andra resurserna. I fall vi inte har någon autentiserings infrastruktur, så används dessa primära autentiseringsuppgifter för att få tillgång till den lokala maskinen och dess säkerhets databas och resurser. Men ifall det finns en autentiserings infrastruktur används dessa primära autentiseringsuppgifter som domän autentiseringsuppgifter. Då användaren vill få tillgång till någon annan resurs eller applikation som kräver andra autentiseringsuppgifter kommer Samlad Inloggnings klient programmet att hämta de motsvarande autentiseringsuppgifterna från cachet och presenterar dessa till autentiserings auktoritet. Om de presenterade uppgifterna är korrekta kommer användaren att också bli automatiskt autentiserad till de andra resurserna. Därför måste de sekundära domänerna lita på den primära domänen [3], alltså måste en trust finnas mellan domänerna.

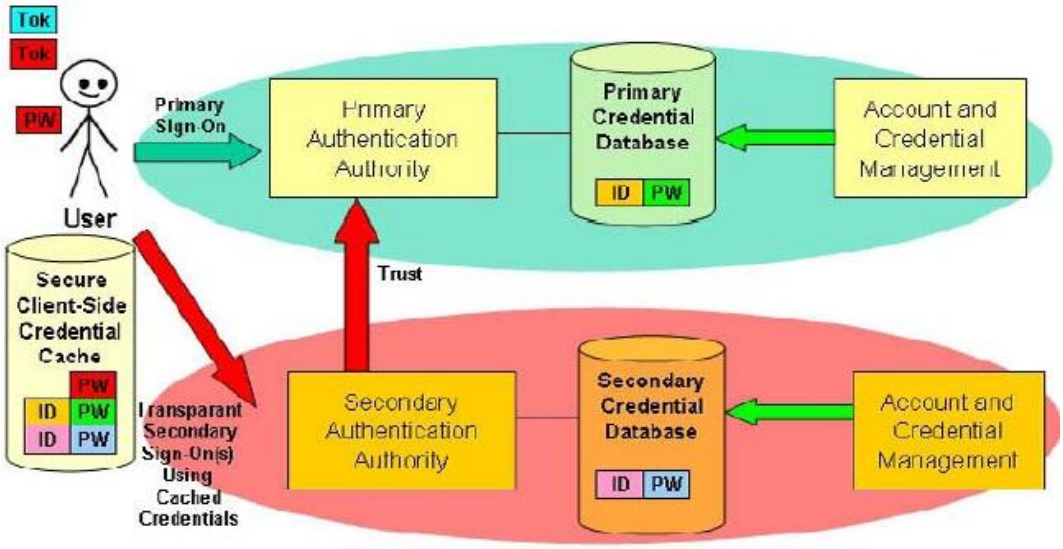


Fig. 5: [11] Secure Client-side Credential Caching Samlad Inloggnings

I denna Samlad Inloggnings arkitektur är säker förvaring av identitetsuppgifterna det essentiella för att skydda tillgången till affärskritiska applikationer eller data. Detta gäller speciellt bärbara klienter som måste ha en hög data säkerhets nivå [3].

Denna arkitektur har lite flexibilitet då all autentiseringsdata är sparad i klient autentiseringsuppgifts cachet i det lokala maskinet. Om en användare försöker autentisera sig genom en annan maskin så kommer det att orsaka autentiserings problem då autentiserings uppgifter inte finns i den ifråga varande maskinen [4].

4.2.2 Secure Server side Credential Cashing Samlad Inloggning

Figur 4 illusterar hur en Secure Server side Credential Cashing Samlad Inloggnings lösning fungerar. I detta fall sparas autentiseringsuppgifterna i en central arkiv på en server istället för att dessa uppgifter skulle sparas på en lokal maskins säkerhets databas. I denna arkitektur finns det ett så kallad huvud autentiseringsuppgiftsdatabas som innehåller en kartläggning av användarens primära- och sekundära autentiseringsuppgifter [3].

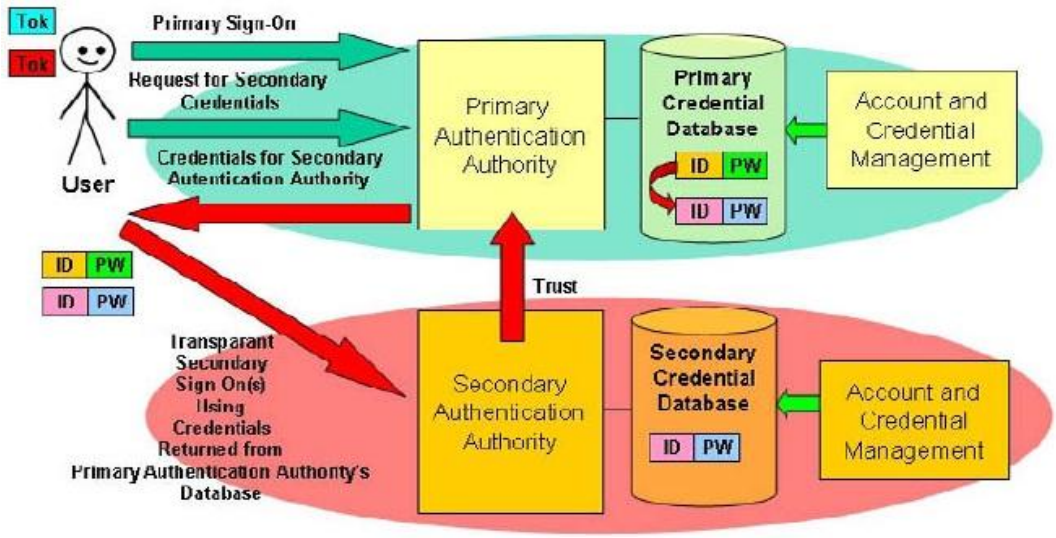


Fig. 6: [11] Secure Server side Credential Cashing Samlad Inloggning

I denna arkitetkur kommer användaren att först autentisera sig till den primära autentiserings auktoriteten genom att använda sig av sina primära autentiseringsuppgifter och en förutbestämd autentiserings protokoll. Den Samlad Inloggnings mjukvaran kommer att antingen ge användaren en lista med tillgängliga applikationer eller överför en säkrad fil till användarens maskin som innehåller användarens sekundära autentiseringsuppgifter. Denna säkrade fil sparas i lokala klienten för resten av inloggningstiden och kommer att förstöras då användaren loggar ut sig från sessionen. Vid behov hämtas autentiserings uppgifter automatiskt från denna säkrade fil och presenteras till autentiserings auktoriteten. Om det igen skickas en lista på tillgängliga applikationer kommer Samlad Inloggnings lösningen att först kommunicera med den primära autentiserings auktoriteten för att hämta de motsvarande autentiseringsuppgifterna, dessa är sedan vidarebefordrade till användaren på ett skyddat sätt. Server side caching medför högre säkerhet emedan autentiseringsuppgifterna sparas endast temporärt under inloggningstiden.

4.2.3 Identitetsuppgifts synkronisering

Identitetsuppgifts synkronisering anses inte vara en sann Samlad Inloggnings lösning. Detta på grund av att användaren hamnar fortfarande att ange sina autentiseringsuppgifter till varje autentiserings auktoritet. Figur 7 illustrerar hur en identitetsuppgifts synkroniserings lösning fungerar.

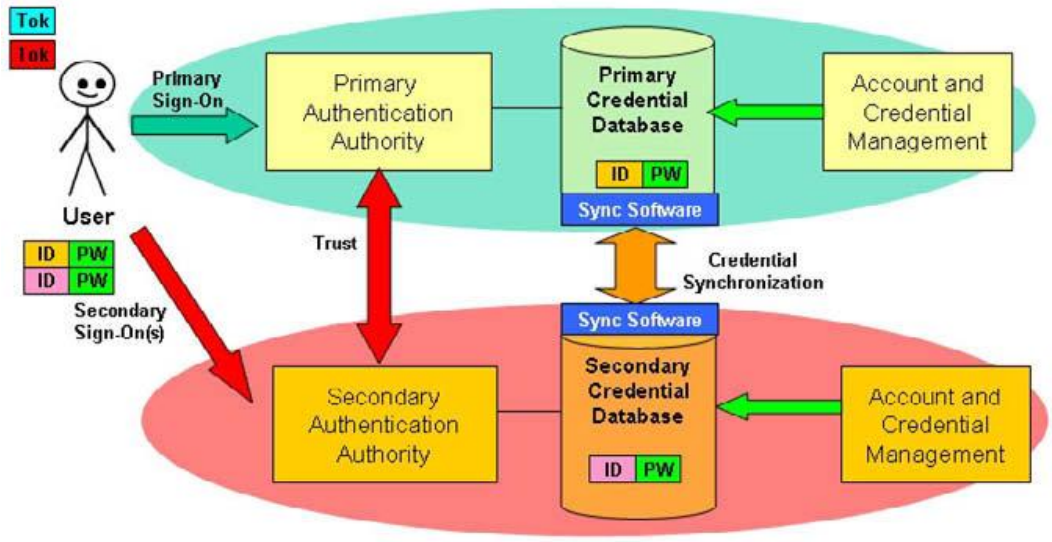


Fig. 7: [11] Identitetsuppgifts synkronisering

En användare kan ha flera olika autentiseringsuppgifter, men på grund av Identitetsuppgifts synkroniserings lösningen kommer autentiseringsuppgifterna att hållas identiska genom autentiseringsuppgifts synkroniserings mekanismen. Identitetsuppgifts synkroniserings lösningar använder sig av ett enda huvudidentitetsdatabas vilket kan gemensamt användas av administratörer för att uppdatera användarnas Identitetsuppgifter.

Teknologin bakom Identitetsuppgifts synkronisering är inte lika lätt som Figur 7 får det att se ut. Ett problem är på vilket sätt identitetsuppgifterna sparas i identitetsuppgiftsdatabasen av de olika autentiseringstillhandhållare. Identitetsuppgifterna sparas oftast i hash format. Detta medför att det är omöjligt att lista ut lösenordet från denna format. Stor del av autentiserings tillhandhållare använder sig av olika hash format, därför går det inte att synkronisera databaserna rakt upp och ner. Identitetsuppgifterna kan endast uppdateras då de skapas eller uppdateras av användaren eller administratören[3, 6, 11].

5. Jämförelse av dessa arkitekturer

Markör baserad Samlad Inloggning.

Fördelar:

Ett enda uppsättning av användar autentiseringsuppgifter underlättar användarnas och administrationens liv.

Nackdelar:

Autentiserings infrastrukturen måste vara i ett homogent omgivning.

Använder sig av symmetrisk kryptografi.

PKI baserad Samlad Inloggning.

Fördelar:

Ett enda uppsättning av användar autentiseringsuppgifter underlättar användarnas och administrationens liv.

Använder sig av asymmetrisk kryptografi.

Nackdelar:

Kan bara hantera en uppsättning av identitetsuppgifter

Certifikat validerings logik är komplext vilket innebär att det krävs mycket bearbetning på klient sidan.

Autentiserings infrastruktur delen måste vara i en homogen omgivning.

Secure Client Side Credential Caching Samlad Inloggning

Fördelar:

Kan hantera flera olika uppsättningar av identitetsuppgifter.

Kräver inte en homogen autentiserings omgivning.

Nackdelar:

Kräver en skyddad klient sidig identitetuppgifts cache.

Flera uppsättningar av identitetsuppgifter försvårar användares och administrationen liv.

Secure Server side Credential Cashing Samlad Inloggning

Fördelar:

Kan hantera flera olika uppsättningar av identitetsuppgifter.

Kräver inte en homogen autentiserings omgivning.

Nackdelar:

Kräver identitetsuppgifts synkroniserings mekanism.

Flera uppsättningar av identitetsuppgifter försvårar användares och administrationen liv

Kräver extra mjukvara på serverns infrastrukturdel.

Identitetsuppgifts synkronisering

Fördelar:

Kan hantera flera olika uppsättningar av identitetsuppgifter.

Kräver inte en homogen autentiserings omgivning.

Lättare att implementera än traditionell Samlad Inloggning

Nackdelar:

Identitetsuppgifterna hålls identiska på olika plattformer.

”Key to the kingdom” argumentet (om någon utomstående får reda på en lösenord så har den tillgång till hela systemet).

Flera uppsättningar av identitetsuppgifter försvårar användares och administrationen liv

Kräver extra mjukvara på serverns infrastrukturdel.

[6, 11]

6. Avslutning

Från en användares synvinkel så är Samlad Inloggning en verkligen bra lösning. Det underlättar deras arbete genom att de hamnar bara att komma ihåg en enda lösenord för att komma åt alla de data resursser som de behöver under en arbetsdag.

I dessa dagar när datasäkerhet förspråkas så kan det gå lätt på det sättet att en lösning blir oanvändbar på grund av att säkerhetsnivån är för hög. Det som man borde sikta mot är en balans mellan användarvänlighet och säkerhet. Samlad Inloggnings lösningar ger företagen den möjligheten att de kan införa en stark lösenords policy, alltså att användarna skall välja en komplex lösenord som de kommer ihåg. Samt att dessa lösenord skall kombineras med biometriska uppgifter om användaren för att få systemet så säkert som möjligt utan att riskera användarvänligheten.

Det som jag lärt mig under denna avhandling är att det inte finns någon specifik lösning till ett företags krav på Samlad Inloggning. Det finns så många faktorer som måste tas i beaktande, några av dem innebär allmänna säkerhetsprinciper, typ av verksamhet, hur viktiga deras legacy och Webbprogram är och kostnadsstrukturen för IT verksamheten. En grundlig undersökning bör utföras för varje faktor förrän man beslutar sig för något. Det kan även vara så att ett företag inte behöver en Samlad Inloggnings lösning utan att de skulle klara sig med en Identitetsuppgifts synkroniserings lösning även om det inte räknas till ett Samlad Inloggnings lösning.

För tillfället så är Samlad Inloggning den bästa lösningen vi har, fastän den inte är perfekt. Framtids aspekter är ljusa, för att tekniken går framåt och detta innebär att ny säkerhets teknik kommer att införas till Samlad Inloggning.

Litteraturförteckning

[1]: Mark Ciampa, Ph.D., Security + Guide To Network Security Fundamentals, 4 ed., Course Technology CENGAGE Learning, år 2012

[2]: William Stallings, Network Security Essentials Applications and standards, 4 ed., Pearson Education, år 2011

[3]: De Clercq, Jan Grillenmeier, Guido, Windows Security Fundamentals, For Windows 2003 SP1 and R2, Digital Press, år 2006

[4]: Si Xiong (12.02.2013) ”Web Single Sign-On System For WRL Company”, Master of Science Thesis, June 2005 Tillgänglig: http://web.it.kth.se/~johanmon/theses/xiong.pdf

[5]: Zubair Ahmad Khattak, Suziah Sulaiman and Jamalul-lail Ab Manan (29.01.2013), Security, Trust and Privacy (STP) Framework for Federated Single Sign-on Environment, November 2011 Tillgänglig: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6122770

[6]: Andrej Volchkov (28.01.2013)**,** Revisiting Single Sign-On A Pragmatic Approach in a New Context, February 2001, Tillgänglig: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=899932&tag=1

[7]: (28.01.2013) The Open Group, Introduction to Single Sign-On, http://www.opengroup.org/security/sso/sso_intro.htm

[8]: Frederick Chong, Microsoft Corporation, Identity and Access Management, July 2004, Tillgänglig http://msdn.microsoft.com/en-us/library/aa480030.aspx

[9]: Single Sign-On Explained, December 17, 2011, Tillgänglig http://idmdude.com/2011/12/17/single-sign-on-explained/

[10]: Huntington Ventures Ltd. (26.02.2013), SSO Federation, 2006, Tillgänglig: http://www.authenticationworld.com/Single-Sign-On-Authentication/SSOFederation.html

[11]: Jan De Clercq, Security Consultant, HP, 2002 http://link.springer.com/chapter/10.1007%2F3-540-45831-X_4?LI=true#page-1