ZIVVERs aanpak van beveiliging en privacy by design

Hoe zorgt ZIVVER er met versleuteling voor dat berichten alleen gelezen kunnen worden door mensen voor wie ze bestemd zijn?

zivver

1. Transport Layer Security **5**

2. Gebruikersauthenticatie en -aanmelding **6**

2.1 Verificatie van inloggegevens 6

2.2 Twee-factor-authenticatie 6

2.3 Gebruikersaanmelding en toekomstige authenticatie met toegangs- en verversingstokens 8

3. Aanmaak en beheer van publieke en privésleutel voor nieuwe gebruikers **9**

3.1 Een nieuw gebruikersaccount wordt aangemaakt door gebruikersnaam en wachtwoord op te geven 9

3.2 Een nieuw publiek/privé sleutelpaar aanmaken 9

3.3 Publieke sleutel van gebruiker opslaan 9

3.4 Generate derived key based on user password 9

3.5 De privésleutel van de gebruiker coderen 10

3.6 De gecodeerde privésleutel opslaan 10

3.7 Onnodige gegevens opruimen 10

4. Informatie delen met andere gebruikers **11**

4.1 Codesleutel genereren 11

4.2 Symmetrisch encryptie van bestand of bericht 11

4.3 Asymmetric encryption of cipher key 11

5. Ophalen van informatie door gebruikers **12**

Aanmelding en ophalen van afgeleide sleutel 12

Ophalen van informatie 12

6. Informatie delen met gasten **13**

6.1 Gasten informatie en toegangsrechten geven 13

6.2 Een nieuw publiek/privé sleutelpaar aanmaken per gast/gesprek 14

6.3 Publieke sleutel voor bericht- en bestandsversleuteling opslaan 14

6.4 Symmetrische sleutel genereren en de voor het gastgesprek specifieke privésleutel coderen en opslaan 14

6.5 Een e-mailbericht versturen met symmetrische sleutel en gebruikersnaam 15

6.6 Privésleutel verpakken met publieke sleutels die ZIVVER gebruikt voor respons 15

7. Ophalen van informatie door gasten **16**

7.1 Gast klikt op notificatiemail 16

7.2 Gast is ingelogd bij ZIVVER 16

7.3 Gast krijgt toegang tot de informatie 16

8. Hoofdsleutelopties voor organisaties **17**

8.1 Een nieuwe organisatie aanmaken met een beheerder 17

8.2 Een nieuw publiek/privé sleutelpaar aanmaken 17

8.3 De privé-hoofdsleutel inpakken met de beheerderssleutel 17

8.4 De publieke sleutel van de organisatie opslaan 18

8.5 Gebruikers toevoegen aan de organisatie en privésleutels inpakken 18

8.6 Beheerdersrechten toekennen aan collega's 18

8.7 Beheerdersrechten van collega's intrekken 19

8.8 Opnieuw toegang geven tot de berichtengeschiedenis van een gebruiker 19

8.9 Toegang krijgen tot de verzonden en ontvangen informatie van andere gebruikers 20

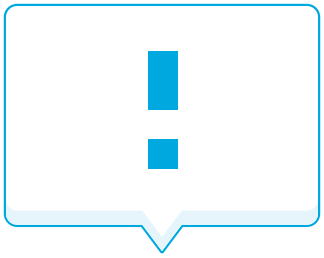
INLEIDING

De meeste mensen denken aan versleuteling of encryptie wanneer ze het hebben over beveiligde communicatie. Ze denken dat hackers hun belangrijkste risico vormen. Maar in de praktijk worden de meeste datalekken veroorzaakt door menselijke fouten, zoals informatie naar de verkeerde persoon sturen.

Als je alleen naar versleuteling kijkt, ben je heel beperkt bezig als het gaat om veilige communicatie. Hiermee beperk je het beveiligingsrisico alleen tijdens verzending en laat je menselijke fouten vóór en ná verzending buiten beschouwing.

Versleuteling is geen wondermiddel, maar het is wel een belangrijk verdedigingsmiddel. Het moet echter op de juiste manier worden ingezet. Toch hebben de meeste mensen weinig kennis van deze materie. In deze paper beschrijven we hoe ZIVVER er met versleuteling voor zorgt dat alleen de (beoogde) afzender en ontvangers de beveiligde berichten kunnen lezen die via ons platform worden verzonden.

Wij van ZIVVER zien het als ons doel gebruikers en organisaties te voorzien van de middelen om communicatie eenvoudig en veilig te maken.



Volgens **Wikipedia** is encryptie "het proces waarbij een bericht of informatie zodanig wordt versleuteld dat alleen de bevoegde partijen er toegang toe hebben. Versleuteling gaat inmenging niet tegen, maar geeft de inhoud niet vrij aan wie het bericht onderschept." En dat is precies wat wij doen. We zorgen er elke dag weer voor dat de gegevens van onze gebruikers, en ook die van hun contactpersonen, veilig zijn. Dat betekent ook dat gevoelige gegevens niet beschikbaar zijn voor ZIVVER. Dit voorkomt dat we zelf een mogelijke bron van datalekken worden, én een interessant doelwit voor hackers.

INLEIDING

Om ervoor te zorgen dat we zelf geen toegang hebben tot gevoelige gegevens, beschikken we niet over de sleutels waarmee je informatie kan ontsleutelen. Dit vereist het gebruik van een versleutelingsalgoritme dat berust op afzonderlijke sleutels voor versleuteling (een publieke sleutel) en ontsleuteling (een geheime of privésleutel). Dit staat ook bekend als asymmetrische versleuteling. Krachtige versleuteling vereist echter het gebruik van sterke sleutels, oftewel: hele lange wachtwoorden. Voor elk bedrijf dat beveiliging serieus neemt, is dit een probleem: de provider kan de sleutel niet opslaan, maar de sleutel is wel nodig wanneer de gebruiker de gegevens wil ontsleutelen. Wat is een haalbare manier om sleutels te beheren, als die procedure tegelijkertijd gebruiksvriendelijk en veilig moet blijven? 'Traditionele' versleutelingsmethoden, zoals PGP, kunnen dit niet bereiken, omdat die berusten op het feit dat gebruikers hun 2048-bits sleutel ergens op een veilige plek bewaren.

ZIVVER vond voor dit probleem een innovatieve en creatieve oplossing.

In deze paper schetsen we de grondbeginselen en werking van onze oplossing in al zijn facetten. We behandelen de volgende onderwerpen:

1. **Transport Layer Security**
2. **Gebruikersauthenticatie en aanmelding**
3. **Aanmaak en beheer van de publieke/privésleutel voor nieuwe gebruikers**
4. **Informatie delen met andere gebruikers**
5. **Gegevens ophalen door gebruikers**
6. **Informatie delen met gasten**
7. **Gegevens ophalen door gasten**
8. **Hoofdsleutelopties voor organisaties**

Alle communicatie tussen clients en het ZIVVER-platform verloopt via een API (Application Programming Interface) op basis van REST-principles. Er wordt gebruik gemaakt van JSON om gegevensobjecten te verzenden. In deze paper zullen we soms verwijzen naar API-aanroepen om duidelijk te maken hoe het platform werkt.

1. Transport Layer Security

Voor alle communicatie tussen clients en servers maakt ZIVVER gebruik van TLS 1.2 (RFC 5246). Het TLS-protocol (Transport Layer Security) biedt privacy en gegevensintegriteit tussen twee communicerende toepassingen. Het is momenteel het meest gebruikte beveiligingsprotocol. Het kan worden gebruikt door webbrowsers en andere toepassingen waarbij gegevens veilig moeten worden uitgewisseld over een netwerk. ZIVVER houdt de ontwikkeling van TLS 1.3 (zie werkconcept) nauwlettend in het oog en zal het toepassen zodra het is afgerond.

2. Gebruikersauthenticatie en -aanmelding

Voor de authenticatie van gebruikers en het verlenen van toegang tot accounts vertrouwt ZIVVER op een combinatie van drie items: de gebruiker die geldige inloggegevens verstrekt in de vorm van een gebruikersnaam, een 'geheim', meestal wachtwoord genoemd, plus een tweede authenticatiefactor.

2.1 Verificatie van inloggegevens

Een gebruiker met ZIVVER-account moet inloggegevens krijgen bij het aanmaken van een account. In ZIVVER is het (primaire) e-mailadres van de gebruiker de gebruikersnaam. Voor het 'geheim' of wachtwoord biedt ZIVVER de volgende opties:

- Een wachtwoord gekozen door de gebruiker. Dit geldt voor gebruikers van persoonlijke accounts, voor beheerders of voor gebruikers van organisaties die geen gebruik maken van eenmalige aanmelding (Single Sign On, SSO) om zich aan te melden bij ZIVVER.
- Een geheim dat wordt geleverd door een SAML 2.0- of OAuth 2.0-compatibele identiteitsprovider (IdP) tijdens de aanmelding. Dit geldt meestal voor gebruikers van organisaties met eenmalige aanmelding (SSO) via bijvoorbeeld een ADFS-setup.

Technisch gezien staat ZIVVER aanmelding toe met een wachtwoord én een of meer geheimen, zoals die door SAML 2.0- of OAuth 2.0-clients aangeleverd worden. Maar voor zakelijke gebruikers kiest de organisatie er meestal voor om alleen aanmelding te ondersteunen via hun SSO-IdP.

ZIVVER gebruikt BCrypt in combinatie met een uniek geheim salt per gebruiker, gegenereerd met een cryptografisch beveiligde pseudotoevalsgenerator (CSPRNG), om hashes van gebruikersgeheimen op te slaan.

Wil je graag meer weten over hashing? Lees dan deze blog.

2.2 Twee-factor-authenticatie

ZIVVER staat gebruikers alleen toe berichten te verzenden als hun account is beveiligd met een extra beveiligingslaag, ook wel twee-factor-authenticatie genoemd. Voor gebruikers met een niet-zakelijk account ondersteunt ZIVVER aanmelding met een extra SMS-code (een op TOTP gebaseerde code verzonden door ZIVVER via een SMS-

provider) of via het gebruik van een authenticator-app die compatibel is met de TOTP-standaard (RFC6238). Gebruikers wordt gevraagd om deze tweede factor in te stellen wanneer ze zich voor de eerste keer aanmelden bij ZIVVER, of elke andere keer dat de gebruiker een bericht probeert samen te stellen of te beantwoorden. Onderdeel van deze tweede factoropzet is de (sterk aangemoedigde) mogelijkheid om tien back-upcodes voor eenmalig gebruik te downloaden als .txt-bestand. Ze kunnen worden ingezet als de gebruiker geen toegang heeft tot de tweede factor (meestal via een telefoon).

Voor zakelijke gebruikers staat ZIVVER inloggen via een SAML-v2-authenticatiecontext toe. Zoals beschreven in paragraaf 2.1 Verificatie van inloggegevens, stelt ZIVVER organisaties in staat SSO te gebruiken via SAML 2.0 om hun gebruikers te authenticeren in ZIVVER. Naast gebruikersnaam en geheim (wachtwoord), biedt ZIVVER organisaties de mogelijkheid om een authenticatiecontext in de SAML-respons door te geven om aan te geven of een gebruiker geverifieerd is, en op welke manier. Hiermee kunnen organisaties voorkomen dat gebruikers zich bij ZIVVER moeten aanmelden met een tweede factor als de IdP van de organisatie de gebruiker al had gevraagd zich aan te melden met een tweede factor.

In dat geval respecteert en erkent ZIVVER de aanvullende authenticatiecontext van IdP. ZIVVER vraagt de gebruiker niet zelf om een tweede factor. In dit scenario komen we de gebruiker tegemoet door hem niet twee keer lastig te vallen, zonder de beveiliging ook maar een moment uit het oog te verliezen.

Gebruikers worden altijd dringend gevraagd te bewijzen dat ze in het bezit zijn van hun account-geconfigureerde tweede factoren. Dit gebeurt zodra ze zich aanmelden bij ZIVVER met geldige inloggegevens op een niet-vertrouwd apparaat. Bij het invoeren van hun tweede factor kunnen ZIVVER-clients gebruikers toestaan een optie te selecteren om het specifieke apparaat te vertrouwen. Als de gebruiker ervoor kiest dit te doen, geeft ZIVVER een unieke apparaatspecifieke sleutel uit. Deze moet bij elke aanmeldingsaanroep door de API-client worden verstrekt als bewijs dat het apparaat vertrouwd wordt.

2.3 Gebruikersaanmelding en toekomstige authenticatie met toegangs- en verversingstokens

Na een succesvolle aanmelding volgens de OAuth 2.0-specificatie (RFC6749), retourneert de API zowel een toegangs- als een verversingstoken in de vorm van een JWT (JSON Web Token/JWT, RFC7519). TDe API-client moet het toegangstoken gebruiken voor toegang tot API-eindpunten waarvoor authenticatie vereist is.

Een toegangstoken heeft een levensduur van enkele minuten. Als het toegangstoken verlopen is, moet je met het verversingstoken een vernieuwd toegangstoken aanvragen.

ThHet verversingstoken is dertig dagen geldig en kan alleen worden vernieuwd door de gebruiker die het wachtwoord opgeeft. Een sessie kan op elk moment worden beëindigd (d.w.z. afgemeld) door het toegangs- of verversingstoken te presenteren aan het herroepingseindpunt (RFC7009). Bestaande JWT's worden van de server uit ongeldig verklaard, omdat de JWT-levensduur niet achteraf kan worden gewijzigd nadat deze aan de klant is verstrekt.

3. Aanmaak en beheer van publieke en privésleutel voor nieuwe gebruikers

De basis van de beveiliging van het ZIVVER-platform wordt gevormd door het gebruik van een cryptosysteem met publieke sleutel in combinatie met een aantal andere methodologieën om zo veel mogelijk te beantwoorden aan het zero-knowledge-doel van ZIVVER. Wanneer een nieuw gebruikersaccount wordt aangemaakt, wordt de volgende procedure gevolgd:

3.1 Een nieuw gebruikersaccount wordt aangemaakt door gebruikersnaam en wachtwoord op te geven

De gebruiker of de organisatie maakt een nieuw account aan door een gebruikersnaam en wachtwoord op te geven als onderdeel van de bijbehorende API-aanroepen. We ondersteunen ook SCIM 2.0 voor accountregistratie.

3.2 Een nieuw publiek/privé sleutelpaar aanmaken

Voor elke nieuwe gebruiker die een ZIVVER-account aanmaakt, wordt een nieuw publiek/privé sleutelpaar gegenereerd met behulp van CSPRNG. Versleuteling vindt plaats met behulp van het RSA-algoritme (Rivest–Shamir–Adleman, 2048 bits).

3.3 Publieke sleutel van gebruiker opslaan

De publieke sleutel van de gebruiker wordt opgeslagen in het ZIVVER-platform om in de toekomst bestanden en berichten mee te versleutelen.

3.4 Generate derived key based on user password

Met het wachtwoord van de gebruiker wordt in combinatie met een uniek 'salt' een hash gegenereerd met behulp van PBKDF2 (Password-Based Key Derivation Function 2). Deze functie ziet er als volgt uit als: Afgeleide sleutel = PBKDF2(PRF, wachtwoord, salt, c, dkLen), waarbij:

i. PRF is een pseudo-random-functie met twee parameters met als uitvoer de lengte hLen (bijvoorbeeld een keyed-HMAC).
ii. Wachtwoord is het wachtwoord van de gebruiker waaruit de afgeleide sleutel wordt gegenereerd.
iii. Salt is een bitreeks, bekend als een cryptografisch salt.
iv. c is het aantal gewenste iteraties, in ons geval 65.536.
v. dkLen is de gewenste lengte van de afgeleide sleutel, in ons geval 128 bits.

3.5 De privésleutel van de gebruiker coderen

We coderen de gegenereerde privésleutel van de gebruiker met behulp van het AES-CTR-algoritme (Advanced Encryption System, met Counter mode, 128 bits),128 bits), waarbij we gebruikmaken van de afgeleide sleutel uit stap 3.4 als symmetrische coderingssleutel.

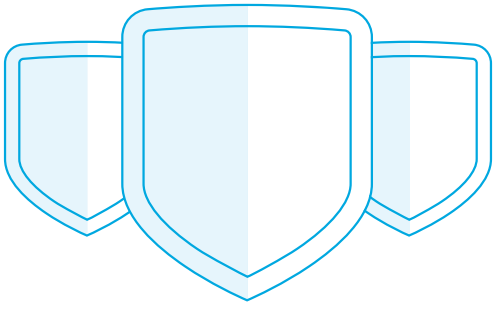
3.6 De gecodeerde privésleutel opslaan

We slaan de gecodeerde privésleutel van de gebruiker op in onze database.

3.7 Onnodige gegevens opruimen

Ten slotte wissen we alle (gevoelige) informatie die niet nodig is uit het geheugen, inclusief het wachtwoord en de privésleutel van de gebruiker.

Bovenstaande stappen worden volledig in het geheugen uitgevoerd op (een van) de ZIVVER-servers. Daarom worden het wachtwoord en de privésleutel van de gebruiker nooit opgeslagen. Deze zijn nooit toegankelijk, voor niemand. Voor de leken onder ons hebben we de volgende analogie. We hebben samen met de gebruiker een lekkere maaltijd bereid en het volledige recept opgeschreven. Maar we kunnen het recept niet opnieuw bereiden zonder de aanwezigheid van de gebruiker, omdat het geheime ingrediënt ontbreekt, namelijk diens wachtwoord... Wat vind je daarvan?



4. Informatie delen met andere gebruikers

Met de hierboven beschreven procedure is het proces van delen van informatie met gebruikers die al een ZIVVER-account hebben, vanuit cryptografisch oogpunt relatief eenvoudig. Dit gebeurt volgens de stappen die in de volgende paragrafen worden beschreven.

4.1 Codesleutel genereren

A new AES-CTR 128 bit cipher key is generated for every file or message.

4.2 Symmetrisch encryptie van bestand of bericht

Het bestand of bericht wordt symmetrisch versleuteld met de gegenereerde sleutel.

4.3 Asymmetric encryption of cipher key

De gegenereerde sleutel wordt (asymmetrisch) gecodeerd met de publieke sleutel van alle gebruikers waarmee de informatie wordt gedeeld.

We versleutelen de bestanden niet rechtstreeks met de publieke sleutels van ontvangers (en afzender), omdat asymmetrische encryptie rekenkundig gezien een zware operatie is. Omdat gebruikers grote bestanden (>1 terabyte) met ZIVVER kunnen delen, zou asymmetrische codering het uiterste vergen van de middelen en bovendien erg tijdrovend zijn. Symmetrische codering (met behulp van AES-CTR) is rekenkundig veel efficiënter. In combinatie met asymmetrische codering van de symmetrische sleutel wordt een hoog beveiligingsniveau gehandhaafd. Voor symmetrische codering hebben we besloten 128-bits sleutels te gebruiken, ook voor rekenkundige performance. AES kent ook 192 en 256 bit-opties, maar deskundigen zijn het erover eens dat het beveiligingsniveau dat wordt verkregen door het gebruik van meer bits, puur theoretisch is. Het wachten is op quantum computing, omdat er veel meer rekenkracht gevergd wordt (zie hier en hier). Zodra quantum computing in beeld komt, zullen we upgraden naar AES-256 of andere nog te ontwikkelen versleutelingsalgoritmen.

5. Ophalen van informatie door gebruikers

Voor gebruikers die al een ZIVVER-account hebben, vindt het ophalen van informatie plaats volgens de stappen die in de volgende paragrafen worden beschreven.

Aanmelding en ophalen van afgeleide sleutel

Bij aankomst op het platform moet iedere gebruiker inloggen volgens de procedure die wordt beschreven in paragraaf 2. Gebruikersauthenticatie. Bij het inloggen moet de API-client namens de gebruiker diens wachtwoord als een van de API-aanroepparameters meegeven. Na een succesvolle aanmelding neemt het ZIVVER-platform het opgegeven wachtwoord en gebruikt dat als een invoerparameter om de afgeleide sleutel opnieuw aan te maken, zoals gegenereerd met PBKDF2, beschreven in stap 4.4. Nadat de afgeleide sleutel opnieuw is aangemaakt, wordt deze als resultaat geretourneerd in de oorspronkelijke login-aanroep.

Ophalen van informatie

De API-client moet de afgeleide sleutel die is verkregen in stap 6.1, leveren als onderdeel van elke toekomstige API-aanroep die gekoppeld is aan het uitvoeren van geverifieerde acties. Bij het ophalen van een bericht of bestand neemt het ZIVVER-platform de afgeleide sleutel van de authenticatiedrager en gebruikt deze om de privésleutel van de gebruiker te decoderen volgens de stappen beschreven in paragraaf 4.5. Vervolgens haalt het platform het symmetrisch versleutelde bericht of bestand op uit het objectenarchief en decodeert de sleutel van het bericht of bestand met behulp van de gedecodeerde privésleutel van de gebruiker. In het geval van een bericht wordt de gedecodeerde instantie geretourneerd als respons op de API-aanroep. In het geval van een bestand retourneert de API een tijdelijke, ondertekende URL waarmee de client (browser) het bestand binnen een korte termijn kan downloaden.

Al het bovenstaande wordt volledig in het geheugen uitgevoerd. Dus noch het wachtwoord, de afgeleide sleutel en de privésleutel, noch het bericht of het bestand worden ooit opgeslagen. Zij zijn dus ook nooit beschikbaar buiten de sessie. Op deze manier kunnen we ervoor zorgen dat informatie alleen leesbaar is voor diegenen aan wie de afzender expliciet toegang heeft gegeven, en niet voor ons of iemand anders...

6. Informatie delen met gasten

Het delen van informatie met gasten ligt veel problematischer: gasten hebben geen account en dus ook geen publiek/privé sleutelpaar om hun informatie mee te versleutelen en te decoderen. Om toch veilig informatie te kunnen delen met gastgebruikers, met inachtneming van het feit dat wij als dienstverlener niet in het bezit mogen zijn van de decoderingssleutels, passen wij de volgende aanpak toe:

6.1 Gasten informatie en toegangsrechten geven

Bij het verzenden van informatie naar een gast moeten ZIVVER-gebruikers a) hun e-mailadres (en eventueel naam) opgeven en b) de toegangsrechten specificeren die de gastgebruiker nodig heeft om het bericht te kunnen lezen. ZIVVER biedt verschillende mogelijke toegangsrechten voor gastgebruikers:

- **SMS-code**: De gebruiker die het bericht verstuurt, kan een mobiel telefoonnummer opgeven waarnaar ZIVVER een op TOTP gebaseerde toegangscode stuurt bij de poging van de gastgebruiker om toegang te krijgen tot het bericht. Om veiligheidsredenen ondersteunt ZIVVER op dit moment geen VoIP-telefoons.
- **Toegangscode**: ZIVVER biedt (leden van) organisaties de mogelijkheid om een organisatiebrede, gastspecifieke toegangscode op te geven die door gebruikers die bij die organisatie behoren, kan worden (her)gebruikt. Denk aan het gebruik van een patiënt- of klantnummer als organisatiebrede toegangscode voor gasten van wie het telefoonnummer onbekend is.
- **E-mailverificatie**: Als de afzender geen telefoonnummer en (organisatiebrede) toegangscode heeft en de gebruiker niet weet hoe hij een (persoonlijke) toegangscode aan de ontvanger moet doorgegeven, kunnen gebruikers met ZIVVER terugvallen op het gebruik van e-mailverificatie. Met deze optie ontvangt de ontvanger een e-mail met een link. Na het klikken op die link ontvangt hij/zij per e-mail een kennisgeving met een tijdelijke toegangslink. Door deze optie te gebruiken verminder je het risico op onderschepping van een normale e-mail aanzienlijk, maar het is niet zo veilig als twee-factor-authenticatie: toegang tot iemands mailbox voldoet om een bericht te lezen.

6.2 Een nieuw publiek/privé sleutelpaar aanmaken per gast/gesprek

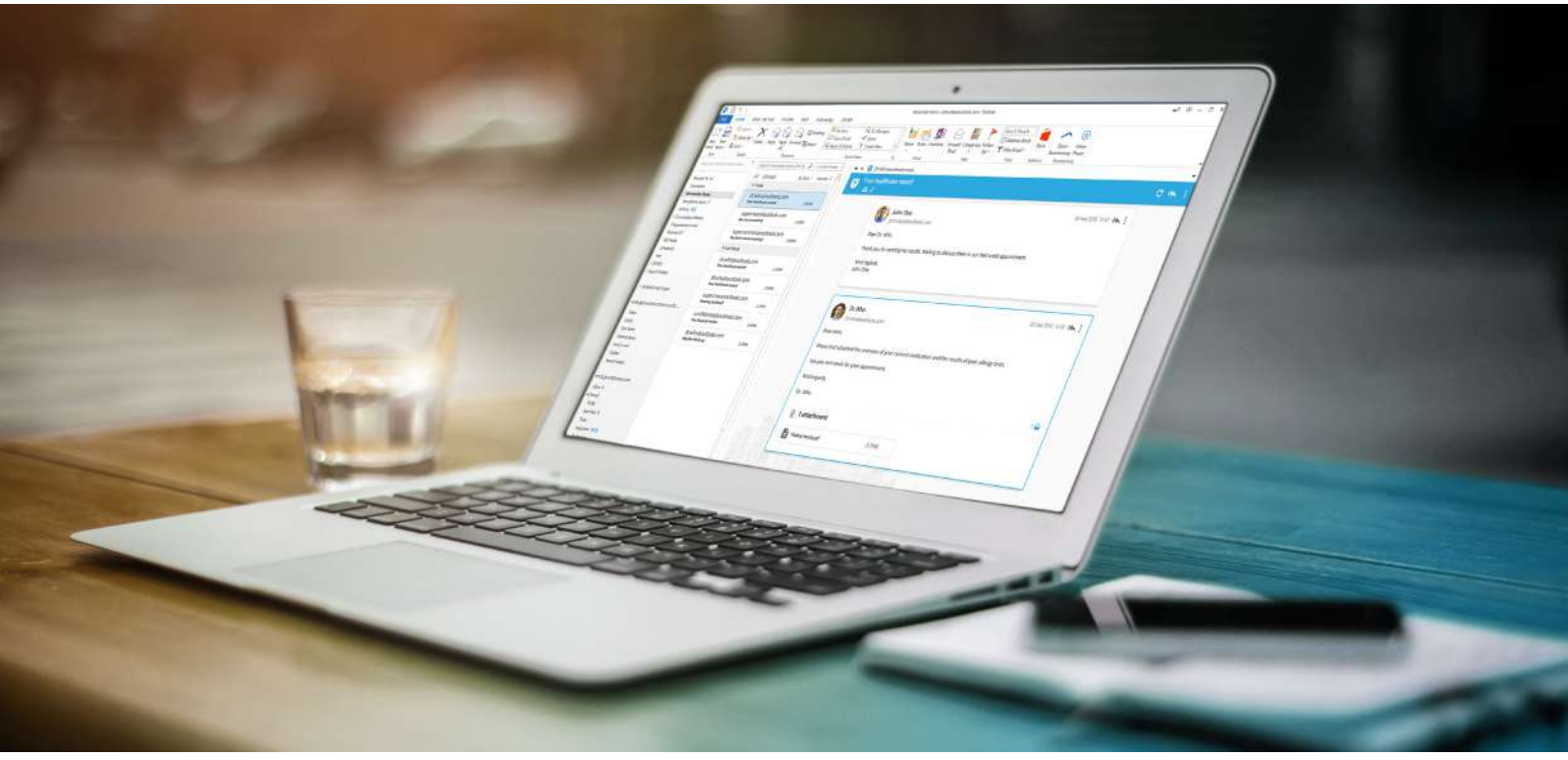
Voor elke nieuwe gast die wordt toegevoegd aan een (nieuw of bestaand) gesprek, wordt een nieuw publiek/privé sleutelpaar aangemaakt voor gebruik van het RSA-algoritme (Rivest–Shamir–Adleman, 2048 bits).

6.3 Publieke sleutel voor bericht- en bestandsversleuteling opslaan

De publieke sleutel van de gebruiker wordt voor later gebruik opgeslagen om bestanden en berichten in hetzelfde gesprek te coderen.

6.4 Symmetrische sleutel genereren en de voor het gastgesprek specifieke privésleutel coderen en opslaan

Vervolgens genereren we een nieuwe AES-CTR 128-bits codesleutel waarmee we de privésleutel voor gastgesprekken coderen. De gecodeerde privésleutel wordt vervolgens opgeslagen. De codesleutel wordt niet opgeslagen, maar in de volgende stap gebruikt.



6.5 Een e-mailbericht versturen met symmetrische sleutel en gebruikersnaam

Gasten sturen we een notificatiemail (opgemaakt en eventueel met logo van de organisatie), waarin ze worden geïnformeerd over het feit dat een specifieke gebruiker hem/haar een beveiligd e-mailbericht gestuurd heeft via ZIVVER. Dit e-mailbericht bevat een link naar het specifieke bericht of gesprek. Een van de URL-parameters in de link bevat de codesleutel waarmee de privésleutel van het gastgesprek kan worden gedecodeerd. Bovendien bevat de notificatiemail informatie over de toegangsrechten die de afzender gekozen heeft (zie paragraaf 7.1).

6.6 Privésleutel verpakken met publieke sleutels die ZIVVER gebruikt voor respons

Tot hier was de verzendprocedure van een bericht naar gasten in wezen niet zo ingewikkeld. Hoe moet je echter omgaan met de situatie waarin een gebruiker een antwoord wil geven in een gesprek waaraan een gast deelneemt? In dat geval zou je de gast nog een notificatiemail willen sturen over het nieuwe bericht. Dit vereist echter ook de privésleutel van de gast voor dat gesprek. Maar die sleutel is niet opgeslagen, dit om ons als serviceprovider mogelijke toegang tot die informatie te ontzeggen. Om dit probleem te tackelen hebben we de volgende oplossing bedacht: de symmetrische sleutel (uit paragraaf 7.4) van het gesprek wordt voor een gast ingepakt (gecodeerd) met de publieke sleutel van elke gebruiker en gast in dat gesprek. Op die manier kan iedereen die mogelijk een antwoord aan het gesprek wil toevoegen, cryptografisch toegang krijgen tot de privésleutel van de betreffende gast, die we vervolgens kunnen opnemen in de notificatiemail, zoals beschreven in paragraaf 7.5.

7. Ophalen van informatie door gasten

Ophalen van informatie (information retrieval) door gasten (gebruikers die geen ZIVVER-account hebben, maar een notificatiemail ontvangen van ZIVVER) werkt volgens de stappen die in de volgende paragrafen worden beschreven.

7.1 Gast klikt op notificatiemail

Zoals beschreven in paragraaf 6.5 ontvangen gastgebruikers een notificatiemail over een nieuw bericht dat via ZIVVER naar hen is verzonden. Zo'n notificatiemail bevat een link waarin de symmetrische sleutel is ingesloten. Met deze sleutel kunnen de privésleutel van het gastgesprek en het e-mailadres waarnaar de e-mail is verzonden, worden gedecodeerd. Als je op de link klikt, word je omgeleid naar de ZIVVER-webapplicatie waar de symmetrische sleutel en het e-mailadres uit de link worden gehaald.

7.2 Gast is ingelogd bij ZIVVER

Met de symmetrische sleutel en het e-mailadres van de link (zie hierboven) maakt de ZIVVER-webapplicatie een login-aanroep naar ZIVVER. Als de verzender een toegangsrecht voor de ontvanger heeft ingesteld (zie paragraaf 7.1), wordt de gast vervolgens gevraagd bewijs te leveren van zijn/haar 'bezit' van het gegeven toegangsrecht.

7.3 Gast krijgt toegang tot de informatie

Als dit gelukt is, wordt de gebruiker aangemeld bij ZIVVER en heeft hij/zij via de ZIVVER-webapplicatie toegang tot het bericht en (eventueel) bijgevoegde bestanden. Hij moet daartoe bij elke API-aanroep de symmetrische sleutel leveren aan het ZIVVER-platform. Decodering van de privésleutel en vervolgens van het bericht en/of de bestanden wordt in het geheugen uitgevoerd en via de client aan de gebruiker aangeboden.

8. Hoofdsleutelopties voor organisaties

Met de hierboven beschreven aanpak is niemand anders dan alleen de gebruiker in bezit van (toegang tot) zijn privésleutel. Dit is echter een ongewenste situatie voor organisaties.

Juridisch gezien is de organisatie waar een gebruiker voor werkt verantwoordelijk voor de handelingen van haar gebruikers en de mogelijke datalekken die zij veroorzaken.

Om volledig verantwoordelijk te kunnen zijn, moeten organisaties de informatie die hun werknemers delen, kunnen controleren en er zelfs toegang toe hebben. Bij andere populaire versleutelingsdiensten is dit meestal niet mogelijk. Hun codering vindt plaats op de client-apparaten met openbare sleutels of ontvangers/deelnemers, en de gegevens zijn niet centraal opgeslagen.

Daarom kunnen organisaties de informatie die werknemers delen nooit volgen, laat staan daar toegang toe hebben. ZIVVER ondervangt deze beperking door middel van een hoofdsleutelconstructie die in de volgende paragrafen wordt beschreven.

8.1 Een nieuwe organisatie aanmaken met een beheerder

Elke gebruiker kan een nieuwe organisatie aanmaken. Hierbij krijgt de gebruiker beheerdersrechten voor die organisatie.

8.2 Een nieuw publiek/privé sleutelpaar aanmaken

Voor elke nieuwe organisatie wordt een nieuw publiek/privé sleutelpaar aangemaakt. Versleuteling vindt plaats met behulp van het RSA-algoritme (Rivest–Shamir–Adleman, 2048 bits).

8.3 De privé-hoofdsleutel inpakken met de beheerderssleutel

De privé-hoofdsleutel van de organisatie wordt ingepakt (gecodeerd) met de (publieke) sleutel van de beheerder die alles aanmaakt.

8.4 De publieke sleutel van de organisatie opslaan

De publieke sleutel van de organisatie wordt opgeslagen. Met deze sleutel kunnen gebruikers toegevoegd worden aan een organisatie (en hun privésleutels ingepakt worden, zie stap hieronder).

Met bovenstaande stappen kunnen beheerders de volgende verrichtingen doen:

8.5 Gebruikers toevoegen aan de organisatie en privésleutels inpakken

Beheerders kunnen gebruikers uitnodigen lid te worden van hun organisatie. Beheerders kunnen ook nieuwe gebruikersaccounts aanmaken en deze automatisch aan hun organisatie toevoegen als ze tot hun organisatie behoren (bijvoorbeeld omdat ze een e-mailadres hebben met een domein dat overeenkomt met een domein dat in het bezit is van de organisatie[1]). Als een gebruiker de uitnodiging accepteert of wanneer een nieuwe gebruiker binnen de organisatie wordt aangemaakt, wordt de symmetrische sleutel waarmee de persoonlijke sleutel kan worden gedecodeerd, ingepakt (gecodeerd) met de hoofdsleutel van de organisatie.

8.6 Beheerdersrechten toekennen aan collega's

Beheerders kunnen beheerdersrechten toekennen aan andere gebruikers binnen hun organisatie. In dat geval wordt de hoofdsleutel van de organisatie gedecodeerd met de privésleutel van de huidige beheerder en wordt de privésleutel van de organisatie vervolgens ingepakt met de (publieke) sleutel van de nieuw toegewezen beheerder.

---

1 Met ZIVVER kunnen beheerders domeinen claimen (bijvoorbeeld hoofddomeinen). Daartoe wordt een verificatiecode verzonden naar een van de volgende e-mailaliassen: admin@, administrator@, hostmaster@, postmaster@ of webmaster@ plus het domein dat wordt geclaimd. Die verificatiecode kan worden ingevoerd in de ZIVVER-webapplicatie (of door een andere API-client) wat ZIVVER als bewijs beschouwt in het bezit te zijn van het opgegeven domein.

8.7 Beheerdersrechten van collega's intrekken

Beheerders kunnen andere beheerders tot gewone gebruikers degraderen. In dat geval wordt de hoofdsleutel van de organisatie, die in de (publieke) sleutel van de betreffende gebruiker ingepakt is, verwijderd.

Met de hoofdsleutelprincipes maakt ZIVVER het beheerders (beter gezegd, gebruikers met een publieke sleutel die een verpakte versie is van de organisatie-hoofdsleutel) op cryptografische wijze mogelijk om toegang te krijgen tot de verzonden of ontvangen informatie van gebruikers. Met deze mogelijkheden kunnen beheerders het volgende doen.

8.8 Opnieuw toegang geven tot de berichtengeschiedenis van een gebruiker

Zoals beschreven in hoofdstuk 4, is het wachtwoord van de gebruiker de sleutel om toegang te krijgen tot zijn/haar privésleutel, waarmee berichten en bestanden gedecodeerd kunnen worden. Als een zakelijke gebruiker zijn wachtwoord kwijt is (bijvoorbeeld wanneer organisaties geen gebruik maken van SSO of de SSO-configuratie verloren is gegaan), kan de gebruiker geen toegang krijgen tot de berichtengeschiedenis. Omdat we geen toegang hebben tot de privésleutel van de gebruiker, kunnen we gebruikers de toegang niet zelf teruggeven. Maar omdat beheerders indirect toegang hebben tot de privésleutel van de gebruiker, kunnen zij dit wel. ZIVVER biedt beheerders dus deze optie, wat in feite op het volgende neerkomt:

i. Decodeer de afgeleide symmetrische sleutel van het oude wachtwoord met behulp van de hoofdsleutel.
ii. Decodeer de afgeleide symmetrische sleutel uit het nieuwe wachtwoord.
iii. Decodeer alle oude privésleutels met de oude afgeleide sleutel.
iv. Codeer al deze privésleutels opnieuw met de afgeleide symmetrische sleutel uit het nieuwe wachtwoord.

Gebruikers hebben vervolgens weer toegang tot hun berichten en bestanden die werden verzonden en ontvangen voordat hun wachtwoord werd gereset.

8.9 Toegang krijgen tot de verzonden en ontvangen informatie van andere gebruikers

Bij een vermoeden van een (ernstig) datalek of frauduleuze handeling door een gebruiker biedt ZIVVER beheerders de mogelijkheid toegang te krijgen tot de berichten of bestanden van een gebruiker door:

i. ze toe te staan zichzelf als gemachtigde toe te voegen aan het account van de gebruiker,

ii. zich aan te melden op het gebruikersaccount door middel van account-delegatie,

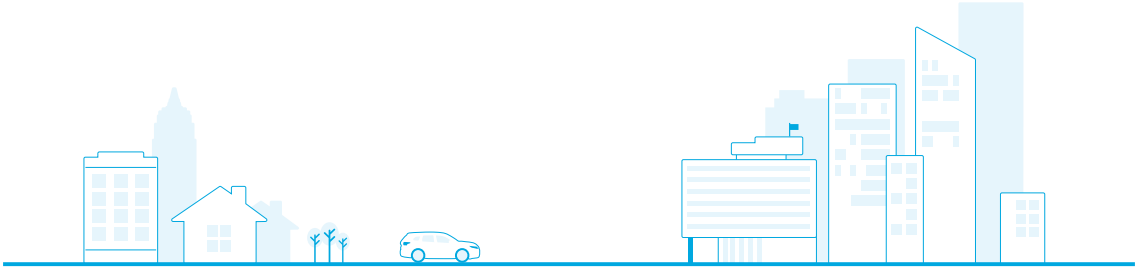
iii. berichten en/of bestanden in te zien met gedelegeerde toegang.

Alle bovenstaande handelingen worden door ZIVVER vastgelegd in een organisatiespecifiek audit- en communicatielogboek dat beschikbaar is voor alle beheerders van die organisatie.

Wil je meer weten?

Neem dan gerust contact op met ons. We staan voor je klaar.

Phone: +31 (0) 85 01 60 555
E-mail: contact@zivver.com
Twitter: @ZIVVER_NL
LinkedIn: /company/zivver



zivver

Secure communication just got simple

Contact

Overschiestraat 186-J

1062 XK Amsterdam

+31 (0) 85 01 60 555

contact@zivver.com