Nationaal Cyber Security Centrum
Ministerie van Veiligheid en Justitie

HTTPS kan een stuk veiliger

Controleer configuraties en pas nieuwe opties toe

Factsheet FS-2014-03
versie 1.1 | 5 november 2014

HTTPS is een veelgebruikt protocol om webverkeer te beschermen tegen partijen die mee willen lezen of het verkeer willen manipuleren. Het configureren van HTTPS luistert nauw: er zijn veel opties en lang niet alle opties zijn veilig.

In deze factsheet worden drie HTTPS-opties uitgelicht die een belangrijke bijdrage kunnen leveren aan het beveiligen van webverkeer. Deze opties zijn aanvullingen op bestaande adviezen over het veilig configureren van HTTPS. Het NCSC adviseert om deze opties toe te passen in al uw HTTPS-configuraties.

Het NCSC adviseert om alle websites die gevoelige gegevens verwerken te beschermen met HTTPS. Wilt u een website beveiligen met HTTPS, dan vindt u configuratieadvies in de ICT-beveiligingsrichtlijnen voor Transport Layer Security en de ICT-beveiligingsrichtlijnen voor webapplicaties van het NCSC.

Aan deze factsheet hebben bijgedragen:
- Belastingdienst
- DefCERT
- Schuberg Philis
- SIDN

De belangrijkste feiten

- HTTPS is een protocol voor het beveiligd versturen van webverkeer tussen een browser en een webserver. HTTPS dient om de vertrouwelijkheid en integriteit van verzonden gegevens te beschermen. Ook kan de identiteit van de eigenaar ermee aangetoond worden.
- Het NCSC adviseert om alle websites die gevoelige gegevens verwerken te beschermen met HTTPS.
- Enkele HTTPS-opties zijn de afgelopen jaren populairder geworden: HSTS, forward secrecy en SHA-2 op certificaten. Deze opties bieden een hoger niveau van bescherming. Het NCSC adviseert om deze opties te gebruiken bij alle websites die met HTTPS beveiligd zijn.
- Gebruikt u HTTPS op uw website, zorg dan voor een veilige configuratie. Gebruik daarvoor de ICT-beveiligingsrichtlijnen voor Transport Layer Security en de ICT-beveiligingsrichtlijnen voor webapplicaties van het NCSC.

De rol van HTTPS

HTTPS is een protocol voor het beveiligd versturen van webverkeer tussen een browser en een webserver. Het verkeer wordt daarbij verstuurd over een internetverbinding of andere niet-vertrouwde verbinding. HTTPS staat voor 'HTTP Secure'. Het protocol komt neer op het versturen van HTTP-verkeer over een verbinding die met Transport Layer Security (TLS)[1] is beveiligd. De verbinding wordt beveiligd met behulp van cryptografie. De veiligheid van een HTTPS-verbinding wordt gewaarborgd door middel van een of twee certificaten.

HTTPS dient om de vertrouwelijkheid en integriteit van gegevens tijdens verzending te beschermen. De vertrouwelijkheid van gegevens is van belang bij het verzenden van gevoelige gegevens, bijvoorbeeld persoonsgegevens. Een buitenstaander moet deze niet in kunnen zien. De integriteit van gegevens is van belang bij het verzenden van instructies, bijvoorbeeld bankoverschrijvingen. Deze moeten authentiek zijn en niet door een buitenstaander te veranderen zijn.

Daarnaast biedt HTTPS voorzieningen voor het vaststellen van de identiteit van de partij waarmee wordt gecommuniceerd. Met behulp van een zogenaamd Extended Validation (EV)-certificaat kan

[1] TLS is de nieuwe naam voor het protocol dat eerder als Secure Sockets Layer (SSL) bekend stond.

een server de identiteit van zijn eigenaar aantonen. Een bezoeker van een internetbankieromgeving met EV-certificaat weet zo bijvoorbeeld niet alleen dat zijn gegevens naar de juiste website worden gestuurd, maar ook dat die website echt van zijn bank is. Sommige browsers ondersteunen deze boodschap visueel, bijvoorbeeld door de adresbalk groen te kleuren. Dit verhoogt het vertrouwen van een bezoeker in de bezochte website.

Wat adviseert het NCSC?

Het NCSC adviseert om alle websites die gevoelige gegevens verwerken te beschermen met HTTPS. Is de vertrouwelijkheid van deze gegevens van belang, of juist de integriteit ervan? Dan is HTTPS een geëigende maatregel, eventueel met een EV-certificaat. Voor bepaalde categorieën websites zal dat al snel het geval zijn:

- websites met formulieren waar bezoekers gevraagd wordt persoonsgegevens in te vullen;
- websites die financiële of medische informatie verwerken;
- websites waarop gebruikers inloggen, bijvoorbeeld met gebruikersnaam en wachtwoord;
- websites die linken naar een website waarop gebruikers moeten inloggen;
- websites waarvan het vertrouwen in de identiteit van de eigenaar van belang is (gebruik hiervoor EV-certificaten).

Gebruikt u HTTPS op uw website, zorg dan voor een veilige configuratie. Het onderliggende beveiligingsprotocol, TLS, kent veel opties. Deze zijn lang niet allemaal veilig. Om een veilige TLS-configuratie te kiezen voor uw website kunt u gebruikmaken van de ICT-beveiligingsrichtlijnen voor Transport Layer Security van het NCSC[5]. Daarnaast kent HTTPS enkele opties die alleen voor webverkeer gelden. Deze komen aan bod in de ICT-beveiligingsrichtlijnen voor webapplicaties van het NCSC[6].

Het Encrypt the Web-initiatief

In 2009 heeft de Electronic Frontier Foundation het Encrypt the Web-initiatief aangekondigd[2]. Dit initiatief vraagt van beheerders om al hun websites met HTTPS te beveiligen. Het verkeer naar die websites is dan minder vatbaar voor onderscheppen of monitoren. Ook is de kans dan kleiner dat beheerders vergeten om een website met gevoelige gegevens van HTTPS te voorzien, aangezien ze deze maatregel op al hun websites toepassen.

Meerdere grote websites bieden inmiddels HTTPS aan. Bekende voorbeelden zijn Google, Facebook, Twitter en Wikipedia. Met behulp van de browserplugin HTTPS Everywhere[3] kunnen gebruikers automatisch naar de HTTPS-versie van een website worden omgeleid als deze beschikbaar is. Deze website moet dan wel in de plugin geregistreerd zijn.

In augustus 2014 heeft Google aangekondigd[4] websites die HTTPS aanbieden hoger in zoekresultaten te zullen vermelden. Op deze manier hoopt Google het gebruik van HTTPS verder te stimuleren.

Enkele HTTPS-opties zijn de afgelopen jaren populairder geworden: HSTS, forward secrecy en SHA-2 op certificaten. Deze opties bieden een hoger niveau van bescherming. Het NCSC adviseert om deze opties te gebruiken bij alle websites die met HTTPS beveiligd zijn. Deze opties komen aan bod in deze factsheet. Ze zijn ook onderdeel van de eerdergenoemde ICT-beveiligingsrichtlijnen van het NCSC.

HTTP Strict Transport Security (HSTS)

Wat HTTP Strict Transport Security (HSTS) is een techniek om browsers op te dragen een website voortaan alleen via HTTPS te bezoeken. De webserver stuurt een extra header mee die de browser deze instructie geeft. Bij elk volgend bezoek aan deze website gedurende een vastgestelde periode zal de browser direct om een HTTPS-versie van de website vragen.

Waarom Als een website HSTS gebruikt, zal een terugkerende bezoeker de website altijd via HTTPS bezoeken. Ook aanvallers die de netwerkverbinding tussen de browser en de webserver kunnen beïnvloeden zijn dan niet in staat om de verbinding terug te zetten van HTTPS naar HTTP. Deze aanval is overigens nog wel mogelijk als een bezoeker voor het eerst een website bezoekt. HSTS beschermt daar niet tegen.

Hoe Om HSTS in te schakelen, dient u deze optie op de webserver in te schakelen. HSTS kent twee opties:

- De webserverbeheerder kan instellen wat de levensduur is van de instructie om de website alleen via HTTPS te bezoeken. Een levensduur van zes maanden tot een jaar is gebruikelijk. Een lange levensduur is voordelig omdat ook infrequente bezoekers dan goed beschermd zijn. Daarmee levert u wel enige flexibiliteit in. Wilt u in de toekomst uw website niet meer via HTTPS aanbieden, dan zult u moeten wachten tot de levensduur van de HSTS-instructie in alle bezoekende browsers is verlopen.
- De webserverbeheerder kan bepalen of de HSTS-instructie ook van toepassing is op subdomeinen. Dan is een bezoek aan de website example.com voldoende om ook de website sub.example.com te beschermen. Gebruik deze instelling alleen als het uw bedoeling is om ook alle websites op subdomeinen van HTTPS te voorzien.

[2] Zie https://www.eff.org/encrypt-the-web.
[3] HTTPS Everywhere is beschikbaar voor meerdere browsers. De plugin is te downloaden via https://www.eff.org/https-everywhere.
[4] Zie http://googleonlinesecurity.blogspot.nl/2014/08/https-as-ranking-signal_6.html.
[5] Zie https://www.ncsc.nl/dienstverlening/expertise-advies/kennisdeling/whitepapers/ict-beveiligingsrichtlijnen-voor-transport-layer-security-tls.html.
[6] Zie https://www.ncsc.nl/dienstverlening/expertise-advies/kennisdeling/whitepapers/ict-beveiligingsrichtlijnen-voor-webapplicaties.html.

Instructies om HSTS in te stellen op uw webserver zijn op internet te vinden. Voor Apache[7], IIS[8] en nginx[9] vindt u verwijzingen in de voetnoten.

Forward secrecy

Wat Forward secrecy is een eigenschap van bepaalde encryptiemethodes voor HTTPS. Gebruikt u een encryptiemethode met forward secrecy, dan is de verbinding niet in te zien voor een aanvaller, ook niet als hij later de beschikking krijgt over de geheime sleutel van het gebruikte certificaat.

Waarom In het volgende scenario is forward secrecy van betekenis. Als een aanvaller toegang tot het netwerk tussen de browser en de webserver heeft, kan hij beveiligd HTTPS-verkeer afluisteren. Weet hij later de hand te leggen op de geheime sleutel van het servercertificaat, dan kan hij het onderschepte HTTPS-verkeer ontsleutelen. De geheime sleutel kan hij bijvoorbeeld achterhalen door een computerhack of een gerechtelijke procedure, desnoods jaren later. Zonder forward secrecy wordt het geheim houden van de geheime sleutel van het servercertificaat steeds belangrijker, omdat de vertrouwelijkheid van steeds meer data er met terugwerkende kracht van afhangt.

Hoe Forward secrecy inschakelen komt neer op het gebruiken van cipher suites voor TLS die 'DHE' of 'ECDHE'[10] in de naam hebben. Een cipher suite is een instelling voor versleutelalgoritmes in TLS. Cipher suites worden ingesteld in de TLS-configuratie van het apparaat waar de TLS-verbinding eindigt. In een eenvoudige opstelling is dat de webserver zelf. In complexere omgevingen kan dat ook de loadbalancer of een externe aanbieder van anti-DDoS-maatregelen zijn.

De server kiest bij het opzetten van een HTTPS-verbinding de cipher suite van zijn hoogste voorkeur die ook ondersteund wordt door de browser. De voorkeur van de server is vastgelegd in de TLS-configuratie. Zet daarom cipher suites met forward secrecy bovenaan deze lijst, zodat ze gekozen worden zodra de client ze ook ondersteunt.

Instructies om forward secrecy in te stellen op verschillende servers zijn op internet te vinden. Omdat deze echter sterk verweven zijn met adviezen over in te stellen cipher suites, vermelden we hier geen links. Raadpleeg de ICT-beveiligingsrichtlijnen voor Transport Layer Security voor advies over in te stellen cipher suites. Raadpleeg de documentatie van uw TLS-programmeerbibliotheek voor instructies om cipher suites in te stellen.

Certificaten met SHA-2

Wat Certificaatleveranciers plaatsen hun digitale handtekening op de hash, de digitale vingerafdruk, van het certificaat. Daardoor kan een bezoeker de authenticiteit van het certificaat controleren. Weet een aanvaller een vals certificaat met dezelfde hash aan te maken, dan kan hij de handtekening van de certificaatleverancier hergebruiken op zijn valse certificaat.

De hash van een certificaat wordt uitgerekend met een hashfunctie. Een hashfunctie is een wiskundige functie die data 'verhaspelt' tot een digitale vingerafdruk. Een veilige hashfunctie maakt het zo goed als onmogelijk om twee verschillende certificaten met dezelfde hash te maken.

Certificaten met SHA-1 als hashfunctie worden vervangen door certificaten met hashfuncties uit de SHA-2-familie: SHA-256, SHA-384 en SHA-512. Certificaten met MD5 als hashfunctie zijn enkele jaren geleden al vervangen. MD5 is de voorloper van SHA-1.

Waarom SHA-1 is een verouderde hashfunctie. Browserbouwers kondigen aan dat ze in de nabije toekomst zullen ophouden certificaten op basis van SHA-1 te accepteren[11,12,13]. Vanaf eind 2014 zullen browsers hun gebruikers gaan waarschuwen als websites certificaten op basis van SHA-1 aanbieden die na 2016 nog geldig zijn. Vanaf 1 januari 2017 zullen certificaten op basis van SHA-1 niet meer geaccepteerd worden.

De reden om SHA-1 uit te faseren is het risico op nagemaakte certificaten. Met een nagemaakt certificaat is het voor een aanvaller met netwerktoegang mogelijk om de vertrouwelijkheid en integriteit van HTTPS-verbindingen te ondermijnen. Voor een andere verouderde hashfunctie, MD5, is aangetoond dat dit in de praktijk mogelijk is[14]. Het valt te verwachten dat dezelfde techniek in de nabije toekomst ook op SHA-1 toepasbaar zal zijn.

Hoe Certificaatleveranciers kiezen zelf op basis van welke hashfunctie ze een certificaat ondertekenen. Gebruikt u momenteel certificaten op basis van SHA-1 (of zelfs MD5), overleg dan met uw certificaatleverancier per wanneer u deze certificaten kunt vervangen door exemplaren op basis van hashfuncties uit de SHA-2-familie. Vraag of nieuwe certificaten alleen nog op basis van hashfuncties uit de SHA-2-familie worden geleverd.

De SHA-2-familie van hashfuncties bevat drie functies die worden gezien als geschikte basis voor handtekeningen op certificaten: SHA-256, SHA-384 en SHA-512. In elk certificaat staat vermeld op basis van welke hashfunctie het ondertekend is.

[7] Instructies voor Apache: https://www.owasp.org/index.php/HTTP_Strict_Transport_Security.
[8] In IIS wordt HSTS ingesteld met een custom header: http://www.iis.net/configreference/system.webserver/httpprotocol/customheaders. De in te stellen header is bijvoorbeeld te vinden op de pagina met instructies voor Apache.
[9] Instructies voor nginx: https://scotthelme.co.uk/setting-up-hsts-in-nginx/.
[10] DHE staat voor Diffie-Hellman Ephemeral. ECDHE staat voor Elliptic Curve Diffie-Hellman Ephemeral. Meer achtergrond bij deze cipher suites en dit advies vindt u in de ICT-beveiligingsrichtlijnen voor Transport Layer Security van het NCSC.

[11] Microsoft geeft aan SHA-1 binnenkort niet meer te accepteren in Windows: http://blogs.technet.com/b/pki/archive/2013/11/12/sha1-deprecation-policy.aspx.
[12] Google geeft aan SHA-1 binnenkort niet meer te accepteren in Google Chrome: http://googleonlinesecurity.blogspot.nl/2014/09/gradually-sunsetting-sha-1.html.
[13] Mozilla geeft aan SHA-1 binnenkort niet meer te accepteren in Mozilla Firefox: https://blog.mozilla.org/security/2014/09/23/phasing-out-certificates-with-sha-1-based-signature-algorithms/.
[14] Zie http://www.win.tue.nl/hashclash/rogue-ca/.

Handelingsperspectief:

1. Het NCSC adviseert om alle websites die gevoelige gegevens verwerken te beschermen met HTTPS. Bedenk daarbij ook hoe gevoelig de gegevens zijn voor degene over wie ze gaan.
2. Zorg voor een goede configuratie van HTTPS. Gebruik daarvoor de ICT-beveiligingsrichtlijnen voor Transport Layer Security en de ICT-beveiligingsrichtlijnen voor webapplicaties van het NCSC.
   - » Het toepassen van HTTP Strict Transport Security (HSTS) is onderdeel van de ICT-beveiligingsrichtlijnen voor webapplicaties.
   - » Het toepassen van forward secrecy is onderdeel van de ICT-beveiligingsrichtlijnen voor Transport Layer Security.
   - » Het gebruiken van certificaten met SHA-2 is onderdeel van certificaatbeheer[15], maar komt ook ter sprake in de ICT-beveiligingsrichtlijnen voor Transport Layer Security.
3. Pas de gekozen HTTPS-configuratie toe op de webserver en aanverwante apparatuur zoals loadbalancers.
4. Van tijd tot tijd veranderen inzichten over welke HTTPS-opties veilig zijn. Dit gebeurt bijvoorbeeld als onderzoekers nieuwe aanvalstechnieken ontdekken. Zorg dat u op de hoogte bent van deze nieuwe inzichten en verwerk deze in uw HTTPS-configuratie. De eerdergenoemde richtlijnen van het NCSC kunnen u daarbij van dienst zijn.

Tot slot

HTTPS is een belangrijke maatregel voor het beschermen van webverkeer dat gevoelige gegevens bevat. In moderne websites is dat steeds vaker het geval. Tegelijkertijd luistert het veilig instellen van HTTPS nauw. HSTS, forward secrecy en certificaten op basis van SHA-2 zijn maatregelen die de veiligheid van webverkeer ten goede komen.

HTTPS wordt op steeds meer websites gebruikt. Dat is begrijpelijk: het is een laagdrempelige maatregel waarmee een website uitstraalt dat de privacy van gebruikers serieus wordt genomen. Als u HTTPS goed instelt, profiteert uiteindelijk iedereen die van uw websites gebruikmaakt.

[15] De factsheet Veilig beheer van digitale certificaten biedt advies over goed certificaatbeheer: https://www.ncsc.nl/dienstverlening/expertise-advies/kennisdeling/factsheets/factsheet-veilig-beheer-van-digitale-certificaten.html.

Uitgave van **Nationaal Cyber Security Centrum**
Turfmarkt 147 | 2511 DP Den Haag
Postbus 117 | 2501 CC Den Haag
www.ncsc.nl | info@ncsc.nl | T 070-751 55 55 | F 070-322 25 37
Publicatienr: FS-2014-03 1.0 | Aan deze informatie kunnen geen rechten worden ontleend.