

UNIFORME BEVEILIGINGSVOORSCHRIFTEN
SECURITY HEADERS

Datum 13-7-2022
Versie 1.0 Definitief
Auteur Edustandaard werkgroep Uniforme Beveiligingsvoorschriften

INHOUDSOPGAVE

1 Inleiding	4
1.1 Achtergrond	4
1.2 Doel	4
1.3 Doelgroep	4
1.4 Samenhang met andere initiatieven	4
1.5 Taken en verantwoordelijkheden	4
1.6 Beheer en doorontwikkeling	4
2 Algemeen	5
2.1 Welke standaarden	5
2.2 Bron voor de security headers	5
2.3 Toetsing	5
2.4 Toepassing van de headers	5
3 Security Headers	6
3.1 Strict-Transport-Security	6
3.2 Content-Security-Policy	7
3.3 X-Permitted-Cross-Domain-Policies	8
3.4 X-XSS-Protection	9
3.5 X-Frame-Options	10
3.6 Public Key Pinning Extension for HTTP (HPKP)	11
3.7 Expect-CT	11
3.8 X-Content-Type-Options	12
3.9 Referrer-Policy	13
3.10 Feature-Policy	13
3.11 Permissions-Policy	13
Bijlage A: Content-Security-Policy	15
Bijlage B: Referrer-Policy	16

Historie

Versie	Auteur	Toelichting	Datum
0.1	Jordy van den Elshout	Opzet en structuur, voor verdere invulling	2 november 2020
0.2	Robert Klein Dennis van Jeveren	Een eerste (basis) uitwerking van de Security Headers	29 januari 2021
0.3	Robert Klein Dennis van Jeveren	Volledige uitwerking van de Security Headers	31 maart 2021
0.4	Jordy van den Elshout	Laatste opmerkingen verwerkt. Versie ter consultatie	3 mei 2021
0.9	Robert Klein	Tips voor toepassing CSP en Referrer-policy toegevoegd in bijlage A en B.	4 januari 2022
1.0	Robert Klein Jordy van den Elshout	Op basis meerdere reviews aanscherpingen en advies toegevoegd voor toepassing. Versie ter goedkeuring.	13 juli 2022

1 INLEIDING

1.1 Achtergrond

Bureau Edustandaard heeft een (advies)verzoek van het 'Ketenregieoverleg PO-VO' (hierna: KRO) gekregen voor de implementatie van de WDO-beveiligingsstandaarden in het onderwijs. Bureau Edustandaard heeft daarvoor de werkgroep Uniforme Beveiligingsvoorschriften (UBV) de opdracht gegeven om WDO-beveiligingsstandaarden in onderwijscontext te plaatsen en uit te werken in voorschriften.

Security Headers vallen (nog) niet onder de WDO, echter is dit thema wel van belang voor de veiligheid van webapplicaties, zoals een website. Kijkend naar internet.nl, die de implementatie van WDO-standaarden inzichtelijk maakt, worden sommige Security Headers wel meegenomen in de test. Wanneer Security Headers niet worden toegepast, kan dat leiden tot kwetsbaarheden in de webapplicatie.

1.2 Doel

Het inzichtelijke maken van 'Security Headers' en de effectiviteit ervan, zodat eigenaren en beheerders van webapplicaties in het onderwijs, deze naar eigen inzicht en situatie kunnen toepassen. Dit heeft als doel om de veiligheid en betrouwbaarheid van webapplicaties in het onderwijs te verbeteren.

1.3 Doelgroep

Deze handreiking is bedoeld voor organisaties die webapplicaties verzorgen en/of beheren voor het onderwijs. Dat geldt voor de hele onderwijssector (PO, VO, MBO en HO).

1.4 Samenhang met andere initiatieven

Deze handreiking is onderdeel van de Uniforme beveiligingsvoorschriften (UBV) voor het onderwijs.

1.5 Taken en verantwoordelijkheden

Het eigenaarschap van dit document is belegd binnen Edustandaard, waar ook andere afspraken binnen het onderwijsdomein worden beheerd. Het beheer en de doorontwikkeling wordt uitgevoerd door de Edustandaard werkgroep UBV.

1.6 Beheer en doorontwikkeling

Binnen Edustandaard wordt periodiek (minimaal eenmaal per jaar) de opzet en de werking van de handreiking besproken met alle relevante stakeholders die vertegenwoordigd zijn in de Standaardisatieraad. Hiertoe wordt input verzameld vanuit de Edustandaard werkgroep Uniforme beveiligingsvoorschriften en vanuit Edu-K.

2 ALGEMEEN

2.1 Welke standaarden

Er zijn talloze (security) headers beschikbaar om een webapplicatie al dan niet veiliger te maken. Deze omschrijving is enigszins omzichtig omdat een juiste inzet en configuratie van de headers noodzakelijk is wil het veiligheid toevoegen. Daarnaast is het de kunst om niet de gebruikerservaring onderuit halen door het toepassen van de headers. Om orde in de chaos te scheppen zijn de relevante headers opgenomen en is aangegeven wat de status is, wat de voors en tegens zijn en waar meer informatie te vinden is.

Helaas is er geen gouden standaard waaruit blijkt welke headers moeten worden geïmplementeerd. Het is aan de verantwoordelijke van de webapplicatie om te bepalen welke headers ingezet worden en in welke mate. Middels dit document wordt een aanzet gegeven om de door de werkgroep aanbevolen headers toch toe te passen.

Verouderde headers blijven met status “deprecated” in het document staan om toch duiding te geven aan de betreffende header. Zo wordt duidelijk waarom een header verouderd is en waardoor deze header vervangen is. Door zowel oude, actuele als nieuwe headers op te nemen ontstaat er een uitputtend overzicht van de beschikbare headers.

Deze handreiking is bedoeld om richting te geven aan de inzet van Security Headers. Er wordt niet ingegaan op hoe de headers geïmplementeerd kunnen of moeten worden. Dit is sterk contextafhankelijk en hier dient per webapplicatie een inschatting voor gemaakt te worden. Waar mogelijk worden wel tips en of verwezen naar best practices, ter ondersteuning van de implementatie.

2.2 Bron voor de security headers

- Toetsing op <https://internet.nl/>
- OWASP Security Headers <https://owasp.org/www-project-secure-headers/>

2.3 Toetsing

De inzet van Security Headers is - naast HSTS - nog altijd vrijblijvend. Dat wil zeggen dat hier niet actief op getoetst wordt. Het is echter uiteraard wel mogelijk een zelftest uit te voeren, zoals dit voor TLS ook al jaren best-practice is. Hiervoor kan gebruik gemaakt worden van bijvoorbeeld <https://securityheaders.com/>. Een beperkte set wordt ook door internet.nl getest, waarbij tevens nadere instructie staat hoe de situatie verbeterd kan worden.

2.4 Toepassing van de headers

In onderstaande overzicht staan alle Security Headers die worden behandeld, verdeeld per toepasselijkheid: de header wel of niet wordt aanbevolen, of in eigen overweging toegepast moet worden. Onder de kop ‘Advies’ van elke header (onder Hoofdstuk 3) staat toegelicht waarom. Dit hangt met name samen met de status van de header. Headers worden standaard niet aanbevolen als deze ‘deprecated’ zijn.

Aanbevolen	Eigen overweging	Niet aanbevolen
<ul style="list-style-type: none">- Strict-Transport-Security (HSTS)- Content-Security-Policy (CSP)- X-Permitted-Cross-Domain-Policies- X-Content-Type-Options- Referrer-policy- X-Frame-Options	<ul style="list-style-type: none">- Permissions-policy	<ul style="list-style-type: none">- X-XSS-Protection- Public Key Pinning Extension for HTTP (HPKP)- Expect-CT- Feature-policy

Overzicht 1 Security Headers per toepasselijkheid

3 SECURITY HEADERS

In dit hoofdstuk worden de Security Headers nader uitgewerkt. Het zijn allemaal headers welke betrekking hebben op http gebaseerd verkeer. Per header is uitgewerkt welk risico deze oplossen dan wel introduceren, wat de impact van toepassing is, waar er meer informatie over de header gelezen kan worden en uiteraard wat de status is van deze header.

3.1 Strict-Transport-Security

Status: Productie

Risico's / toegevoegde waarde

Het toepassen van de HSTS-header volgt logischerwijs uit de verplichting om:

- HTTPS/TLS voor alle overheidswebsites toe te passen
- Kwaliteitseisen t.a.v. encryptie en toepassing hiervan (zoals beschreven in de ICT-Beveiligingsrichtlijnen voor Transport Layer Security)

De HSTS-header voorkomt de mogelijkheid om onversleuteld met een server te communiceren. En daarmee ook een Man-In-The-Middle aanval (MITM-aanval).

Zonder het toepassen van de HSTS-header staat de dienstverlening bloot aan de volgende risico's (allen via een MITM-aanval):

- Gegevens van gebruikers worden gestolen
- Gegevens van gebruikers en aangeboden content kan worden gemanipuleerd
- Gemanipuleerde gegevens vinden hun weg door de keten heen en worden op derden-systemen opgeslagen

De header wordt pas actief in de browser wanneer de bezoeker de website minimaal eenmaal via https heeft bezocht, of wanneer het domein in de preload list is gezet door de website eigenaar. Alle grote browsers ondersteunen deze lijst. Hier is niet eerst een bezoek aan een domeinnaam voor nodig om de HSTS configuratie op te halen, deze is dan reeds aanwezig in de browser. Aanmelden kan via <https://hstspreload.org/>.

Het is aan te bevelen om de parameter max-age in te stellen op 31536000. Dit betekent dat de browser de hsts instelling een jaar lang onthoudt. Hierdoor zal de browser zelfs zonder de header te zien HSTS toch direct toepassen.

De parameter includeSubDomains kan worden opgenomen als het domein en alle subdomeinen van een volledig correcte TLS configuratie zijn voorzien. Belangrijk is om dit eerst te valideren voor alle subdomeinen omdat anders toegang tot webdiensten waar er problemen met certificaten zijn of zelfs TLS niet wordt toegepast, onmogelijk wordt.

Er moet echter wel rekening gehouden worden dat wanneer een certificaat bijv. verloopt een gebruiker een melding krijgt in de browser die niet meer te negeren is. Zonder HSTS kan je alsnog de melding negeren en doorgaan, met HSTS is dit niet mogelijk.

Implementatie impact

Triviaal: er hoeft slechts een header aan webserver configuratie of aan de applicatie code toegevoegd te worden.

Functionele impact

Laag: Communicatie over onversleutelde HTTP verbindingen wordt technisch voorkomen. De kwaliteit van de

TLS verbinding (o.a. toegestane encryptie algoritmes, configuratie, e.d.) valt buiten de scope van deze header. De eindgebruiker merkt alleen verschil wanneer een ongeldig of verlopen certificaat wordt getoond.

Security impact

Hoog: De header voorkomt dat een browser nog terug kan vallen op plain-text communicatie richting de betreffende website of applicatie en voorkomt dat een gebruiker nog akkoord kan gaan wanneer een ongeldig certificaat wordt getoond.

Advies

De HSTS header wordt sterk aanbevolen om toe te passen. Deze biedt een hoge mate van toegevoegde waarde op de beveiliging van de webdienst.

Documentatie

<https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/Strict-Transport-Security>

3.2 Content-Security-Policy

Status: Productie

Risico's / toegevoegde waarde

Deze header biedt een breed scala aan mogelijke bescherming tegen kwaadwillende content op websites. Gezien de mogelijkheden die de header hierin biedt moet er voorzichtig omgegaan worden met het inzetten ervan. CSP kan veel invloed hebben op hoe een browser de content weergeeft. Een verkeerde implementatie kan zelfs zorgen dat de website niet, of slechts deels functioneert. Deze policy heeft een "report-only" modus, zodat de policy geëvalueerd kan worden zonder dat de policy daadwerkelijk zal worden toegepast.

CSP heeft invloed op minimaal de volgende html, css en JavaScript elementen:

- Scripts
- Plugin objecten
- Stylesheets
- Plaatjes
- Frames
- Video/audio
- Fonts
- Forms

Voor elke aanpassing aan de CSP header moet er goed getest worden omdat een kleine aanpassing aan de generieke header, impact heeft op elke pagina van de website of applicatie. Ook als de website of applicatie wordt aangepast.

Voor een veilige implementatie kan gebruik gemaakt worden van de 'Content-Security-Policy-Report-Only' header. Deze header zal afwijkingen van het ingestelde CSP beleid enkel rapporteren, maar niet afdwingen in de browser. Op deze wijze kan men inzichtelijk maken of er problemen zijn met het ingestelde CSP beleid. Indien het beleid geen belemmeringen oplevert kan de 'Content-Security-Policy-Report-Only' header worden vervangen voor de 'Content-Security-Policy' header. Ook het gebruik van de CSP scanner (zie documentatie) kan helpen de Content-Security-Policy header verder in te richten.

In Bijlage A worden tips gegeven voor succesvolle inzet van deze header.

Implementatie impact

Hoog: Een aanpassing aan de header moet altijd grondig getest worden. Tevens moet elke functionele wijziging op de website gecheckt worden tegen de header instelling. Bijvoorbeeld: er is een CSP policy actief welke inladen van externe JavaScripts verbiedt, echter wordt besloten de jQuery scripts te gaan inladen via

een extern CDN. De header zal deze calls naar het CDN nu blokkeren. Dit kan in gevallen tot onverklaarbaar gedrag in de browser leiden waarbij de header al snel over het hoofd gezien wordt. Voor applicaties geldt dat het testen van de gevoerde CSP header vast onderdeel moet worden van de applicatie lifecycle.

Functionele impact

Hoog: Een goed ingestelde CSP header levert geen hinder voor de functionaliteit op, echter om daar te komen zal er grondig getest moeten worden en tevens moet het testen hiervan vast onderdeel worden van de applicatie lifecycle. Deze header is geen eenmalig in te stellen element in de beveiliging. Begin bij het uitrollen van de header met de header "Content-Security-Policy-Report-Only" om zo het effect te monitoren zonder de configuratie direct af te dwingen.

Security impact

Hoog: Naast dat de header functioneel en qua implementatie het nodige vergt levert het ook wat op. De header levert een enorme reductie in de kans dat malafide content ingeladen kan worden. Alleen vertrouwde bronnen mogen worden gebruikt om content in de browser te laden. Ook al wordt een website gecompromitteerd, kunnen gebruikers verkeerde content uploaden of proberen derden de website middels frames te misbruiken, de CSP header zorgt dat de browser dit blokkeert. Hierdoor is de gebruiker in hoge mate beschermd tegen onbedoeld gebruik van de website of applicatie.

Belangrijk hierbij is wel om het gebruik van wildcards (*) te minimaliseren. Een algemene wildcard, welke dus alle domeinnamen toestaat wordt sterk afgeraden, dit voegt geen veiligheid toe. Het toepassen van een wildcard op subdomeinen (*.domein.tld) moet alleen als laatste optie overwogen worden. De CSP levert alleen echte beveiligingsvoordelen als domeinen volledig uitgeschreven worden.

Reporting

Om implementatie makkelijker te maken kan deze header overtredingen rapporteren naar een endpoint. Deze rapporten kunnen worden geanalyseerd om zo een solide policy vast te stellen. Het endpoint kan onder eigen beheer staan, of er kan gebruik gemaakt worden van een commerciële partij die de reporting verzorgt. Zowel de 'report-only' header als de 'Content-Security-Policy' header ondersteunen reporting. In het laatste geval kan dit nuttig zijn om blijvend te kunnen monitoren.

Let bij de reporting functie op waar de reports naartoe gestuurd worden. Er kunnen in beperkte mate privacygevoelige gegevens verstuurd worden zoals de aangeroepen url, referrer en user-agent. IP adressen, cookies en http post content van gebruikers worden niet verstuurd. In enkele gevallen kan bijvoorbeeld de url echter toch gevoelige gegevens bevatten.

Advies

Het toepassen van de CSP header wordt aanbevolen, met name waar het gaat om het inladen van risicovolle content als scripts en plugins. Mocht de later in dit document aangehaalde X-Frame-Options header niet toegepast worden dan is het sterk de aanbeveling om ook frames configuratie op te nemen in de CSP header.

Documentatie

<https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/Content-Security-Policy>

<https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/Content-Security-Policy-Report-Only>

<https://content-security-policy.com/>

3.3 X-Permitted-Cross-Domain-Policies

Status: Productie

Risico's / toegevoegde waarde

Er wordt een cross-domain policy xml toegepast voor webobjecten, zoals bijv. Adobe Flash en Adobe PDF waarmee wordt aangegeven of data tussen domeinen verstuurd mag worden. Middels deze header kan aangegeven welk cross-domain policy bestand gehanteerd mag worden.

De header heeft vooral toegevoegde waarde gehad bij het gebruik van Adobe Flash animaties. Nu dit zo goed als verdwenen is omdat support voor Flash in alle browsers is gestopt zal de relevantie van deze header afnemen.

Implementatie impact

Triviaal: er hoeft slechts een header aan webserver configuratie toegevoegd te worden

Functionele impact

Nihil: Verandert aan de functionaliteit van de website niets

Security impact

Laag: Op websites waar gebruikers bestanden kunnen uploaden, content kunnen aanpassen, kan deze header een voordeel opleveren omdat de policy globaal gezet wordt en een malafide crossdomain.xml dus zal worden genegeerd.

Advies

De toepassing van deze header wordt aanbevolen. Als er gebruik gemaakt wordt van crossdomain policies en het xml bestand kan niet in de webroot geplaatst worden, dan is configuratie van deze header noodzakelijk. In situaties waarbij er überhaupt geen noodzaak is tot cross domain policies dan zou deze header ingesteld moeten worden op 'none'.

Documentatie

<https://www.adobe.com/devnet-docs/acrobatetk/tools/AppSec/xdomain.html>

[Cross-domain policy file specification \(adobe.com\)](#)

3.4 X-XSS-Protection

Status: Deprecated

Risico's / toegevoegde waarde

De X-XSS-Protection header is een tijdelijke HTTP security header die Cross Site Scripting aanvallen kon voorkomen. Bij ingeschakelde Cross Site Scripting bescherming controleert de browser uitgaande HTTP requests op potentieel kwaadaardige strings. Vervolgens wordt naar hetzelfde string patroon gezocht in het antwoord van de server. Indien het patroon gevonden wordt in het antwoord, dan wordt de conclusie getrokken dat er een XSS aanval plaats vindt.

Er zijn verschillende problemen met deze header:

1. Legitieme scripts kunnen geblokkeerd worden
2. Er zijn veel manieren om deze controles te omzeilen.

<https://blog.innerht.ml/the-misunderstood-x-xss-protection/>

De header wordt niet meer toegepast in moderne browsers en is vervangen door de Content-Security-Policy header.

Implementatie impact

Triviaal: er hoeft slecht een header aan de webserver configuratie of aan de applicatie code toegevoegd te worden.

Functionele impact

Deze header is opgevolgd door de Content-Security-Policy header. Het advies is om de functionaliteit van deze header uit te schakelen middels *X-XSS-Protection: 0* zodat oude versies van browsers deze standaard waarde niet zullen toepassen. Gebruik voort de Content-Security-Policy header.

Security impact

Laag: de meeste browsers hebben de ondersteuning voor deze header uitgezet.

Advies

De toepassing van deze header wordt niet aanbevolen. Browsers hebben ondersteuning van deze header uitgebouwd waardoor deze effectief niets meer toevoegt. Zie https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/X-XSS-Protection#browser_compatibility voor informatie over ondersteuning van deze header.

Documentatie

<https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/X-XSS-Protection>

3.5 X-Frame-Options

Status: Productie, maar kan vervangen worden door CSP “frame-ancestors”

Risico's / toegevoegde waarde

De header beschermt tegen zgn. clickjacking aanvallen, waarmee webpagina's voor de gebruiker onzichtbaar in een iframe geladen worden. Een kwaadwillende kan doordat de webpagina in 'zijn' iframe geladen wordt de pagina clicks afvangen en omleiden. De gebruiker denkt met de click een legitieme actie uit te voeren op webpagina, maar voert feitelijk een actie bij de kwaadwillende uit. De header geeft aan de browser aan in hoeverre de webpagina via een iframe gelaten mag worden.

De header is inmiddels opgenomen als onderdeel van de Content Security Policy waarmee deze header gaat komen te vervallen. Oudere browsers (IE11 en ouder, Safari 9.2 IOS) ondersteunen dit Content Security Policy onderdeel niet. Het verdient voorlopig de aanbeveling deze header te blijven voeren, naast de Content Security Policy header zodat ook oudere browsers hiervan gebruik kunnen maken. Wanneer deze header en CSP “frame-ancestors” actief zijn dan heeft CSP “frame-ancestors” voorrang boven X-Frame-Options.

Implementatie impact

Triviaal: er hoeft slecht een header aan de webserver configuratie of aan de applicatie code toegevoegd te worden.

Functionele impact

Middel: De toepassing van frames is heden ten dage beperkt. De noodzaak om een website van derden middels frames in te laden is meestal niet meer aanwezig en zou wanneer wel nodig anders opgelost moeten worden. Controleer voor toepassing van de header of de website wellicht onbedoeld niet toch ergens als onderdeel van een frame is opgenomen. Bijvoorbeeld door opvallende terugkerende referrers in de logging na te lopen.

Security impact

Groot: De gebruiker is beschermd tegen clickjacking aanvallen.

Belangrijk is wel dat de waarde van de header wordt ingesteld op DENY om de website volledig te beschermen tegen inbedding in een frame. Mocht het framen van de website toch noodzakelijk zijn, sta dan alleen toe dat dit gebeurt binnen dezelfde website, niet via een andere website middels waarde SAMEORIGIN.

Advies

Het is voornamelijk aanbevolen deze header te blijven voeren. Gezien deze header in een overgangsfase zit van status productie - op het moment van schrijven - naar status deprecated over een bepaalde tijd, verdient het

de aanbeveling de ontwikkelingen in de gaten te houden. Voor het beschermen van legacy browsers zal echter niets veranderen, deze blijven de header uiteraard respecteren.

Documentatie

<https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/X-Frame-Options>

<https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/Content-Security-Policy/frame-ancestors>

3.6 Public Key Pinning Extension for HTTP (HPKP)

Status: Deprecated

Risico's / toegevoegde waarde

De header beschermt tegen een Man-In-The-Middle aanval (MITM-aanval) door aan te geven welke SSL certificaten gebruikt mogen worden.

De header is inmiddels verouderd en is opgevolgd door Expect-CT. De meeste browsers bieden geen ondersteuning meer voor deze header.

Advies

De toepassing van deze header is niet aanbevolen. Ook niet met het oog op oude browsers.

Documentatie

https://developer.mozilla.org/en-US/docs/Web/HTTP/Public_Key_Pinning

3.7 Expect-CT

Status: Deprecated per juni 2021

Risico's / toegevoegde waarde

De header instrueert de browser om elk aangeboden certificaat van de website te checken in de publieke certificaat logs, de "Certificate Transparency logs". Elk publiek uitgegeven certificaat komt voor in deze logs.

De header wordt inmiddels vervangen door een definitieve oplossing binnen de uitgegeven certificaten. Hiertoe wordt een SCT (Signed Certificated Timestamp) veld toegepast in elk uitgegeven publiek certificaat. Per juni 2021 zijn alle publieke certificaten voorzien van een SCT. Hiermee is validatie in de Certificate Transparency logging mogelijk. De header raakt hierdoor zijn functie kwijt en de browsers bouwen de ondersteuning hiervoor af.

Om de uitgevende CA's te beperken tot alleen de jou bekende leveranciers is het mogelijk in DNS het CAA record in te stellen. Dit record moet door de CA's gerespecteerd worden en verkleint daarmee de kans dat vreemde CA's certificaten kunnen uitgeven voor jouw domeinen.

Implementatie impact

Triviaal: er hoeft slecht een header aan de webserver configuratie of aan de applicatie code toegevoegd te worden.

Functionele impact

Nihil: Functioneel verandert er niets

Security impact

Groot: De gebruiker is beschermd tegen een gecompliceerde Man-In-The-Middle aanval (MITM-aanval) waarbij ongeldige, of frauduleuze certificaten worden gebruikt.

Reporting

Deze header ondersteunt het rapporteren van overtredingen naar een endpoint. Het endpoint kan onder eigen beheer staan, of er kan gebruik gemaakt worden van een commerciële partij die de reporting verzorgt.

Advies

Het wordt niet aanbevolen deze header toe te passen. Deze header heeft nog maar relatief kort bestaan en heeft derhalve maar beperkte browserondersteuning. Daarbij komt dat de header alweer uitgefaseerd wordt.

Documentatie

<https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/Expect-CT>

<https://certificate.transparency.dev/howctworks/>

https://en.wikipedia.org/wiki/DNS_Certification_Authority_Authorization

3.8 X-Content-Type-Options

Status: Productie

Risico's / toegevoegde waarde

De header beschermt de browser tegen het zelfstandig bepalen welk MIME-type een gedownload bestand is waardoor alleen nog expliciet afgegaan wordt op het Content-Type dat in de HTTP header is meegegeven. De webserver moet voor elk MIME-type dan ook de juiste content-type header meegeven. Wanneer bijvoorbeeld een html pagina door de webserver niet content-type text/html heeft meegekregen, dan zal de webpagina een downloadbaar zijn ipv een door de browser gerenderde pagina.

Nb. De header wordt alleen toegepast op de "style" en "script" html elementen in een webpagina.

Implementatie impact

Triviaal: er hoeft slechts een header aan de webserver configuratie of aan de applicatie code toegevoegd te worden.

Functionele impact

Nihil: Als de webserver de CSS en JavaScript bestanden met het juiste content-type aan de browser aanlevert (wat normaliter door een webserver ingebouwde functionaliteit is) zou hier functioneel geen voor- noch nadeel van bemerkt moeten worden.

Security impact

Middel: Op websites waar gebruikers bestanden kunnen uploaden, content kunnen aanpassen, etc. kan deze header helpen beschermen tegen het laden van content die door de browser bijvoorbeeld als script uitgevoerd kunnen worden. Bijvoorbeeld een bestand dat geplaatst wordt met MIME-type text/plain zou door een browser geïnterpreteerd kunnen worden als text/javascript.

Aanbevolen instelling voor deze header is nosniff.

Advies

Het is aanbevolen deze header te gebruiken. De header heeft duidelijk toegevoegde waarde op interactieve websites, voor de beveiliging van de gebruikers.

Documentatie

<https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/X-Content-Type-Options>

3.9 Referrer-Policy

Status: Productie

Risico's / toegevoegde waarde

De header beschermt tegen het lekken van informatie over waar je vandaan bent gekomen als je op een link hebt geklikt. Standaard gedrag van een browser is om, wanneer je doorklikt op een link, de url van waar je op deze link hebt geklikt mee te sturen. Hierdoor komt de website waar je dan naartoe gaat te weten waar jij vandaan gekomen bent. Deze header blokkeert dit gedrag. De header zorgt ervoor dat doorkliks vanaf jouw website niet langer de weblocatie meesturen naar de externe website waar een gebruiker op doorklikt.

In Bijlage B worden tips gegeven voor succesvolle inzet van deze header.

Implementatie impact

Triviaal: er hoeft slechts een header aan de webserver configuratie of aan de applicatie code toegevoegd te worden. Het advies is om ten minste gebruik te maken van "same-origin", zodat de locatie binnen het domein blijft.

Functionele impact

Middel: Deze header richt zich met name op de privacygevoeligheid van data die in een URL aanwezig kan zijn. Nadeel kan zijn dat externe websites niet meer kunnen achterhalen hoe gebruikers op hun website terecht zijn gekomen. Dit kan nodig zijn om statistische redenen maar ook i.v.m. troubleshooting.

Security impact

Middel: Het zorgt ervoor dat gevoelige informatie uit een URL niet wordt doorgestuurd naar andere websites. Tevens kan het de privacy van bezoekers van de website verhogen mits goed ingesteld, zodat andere websites niet kunnen zien waar de bezoekers vandaan zijn gekomen.

Aanbevolen instelling voor deze header is same-origin. Hiermee wordt een referrer alleen nog meegegeven binnen dezelfde website. Links naar buiten de website geven de referrer niet meer mee.

Advies

Het is aanbevolen deze header te gebruiken. Deze header spitst zich toe op de privacy van gebruikers en heeft daarop een toegevoegde waarde.

Documentatie

<https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/Referrer-Policy>

3.10 Feature-Policy

Zie Permissions-Policy. Feature-Policy was de oude naamgeving van deze header.

3.11 Permissions-Policy

Status: Experimenteel

Risico's / toegevoegde waarde

Met deze header kan een breed scala aan functionaliteit en API's in de browser worden in- of uitgeschakeld. Deze header bevindt zich nog in de 'draft' fase. Hierdoor kan de functionaliteit en het gedrag wat de header beoogt te bereiken wisselen. Het is niet aanbevolen deze header nu al toe te passen in productie.

Implementatie impact

Middel: Met deze header kunnen allerlei functies worden beperkt. Er dient grondig getest te worden of een functie in gebruik is, alvorens deze middels de Permissions-Policy te beperken.

Functionele impact

Hoog: Een goed ingestelde Permissions-Policy header levert geen hinder voor de functionaliteit op, echter om daar te komen zal er grondig getest moeten worden en tevens moet het testen hiervan vast onderdeel worden van de applicatie lifecycle. Deze header is geen eenmalig in te stellen element in de beveiliging.

Security impact

Hoog: Het vergt enige tijd om de header functioneel goed in te regelen, maar hiermee kan worden voorkomen dat functionaliteit welke zelf niet gebruikt wordt niet aan te bieden. Aanvallen welke misbruik willen maken van uitgeschakelde functionaliteit zullen hierdoor niet slagen.

Reporting

Deze header ondersteunt het rapporteren van overtredingen naar een endpoint. Het endpoint kan onder eigen beheer staan, of er kan gebruikgemaakt worden van een commerciële partij die de reporting verzorgt.

Advies

Het is ter eigen overweging om deze header wel of niet te gebruiken in een productieomgeving. Deze header is nog in ontwikkeling en ondersteuning ervan is nog wisselend. Daarnaast kan de uitwerking van configuraties nog verschillen in deze fase. Gezien op moment van schrijven van deze handreiking de impact nog niet voldoende overzien kan worden is er geen expliciet advies te geven of deze header wel of niet nu al toegepast kan of moet worden.

Het verdient echter wel de aanbeveling om alvast kennis op te doen van deze header en er in een testomgeving mee te experimenteren. De verwachting is dat de header breed toegepast zal gaan worden.

Documentatie

<https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/Feature-Policy>

<https://www.w3.org/TR/permissions-policy-1>

BIJLAGE A: CONTENT-SECURITY-POLICY

De CSP header is een http header met een uitgebreid palet aan opties en functionaliteiten. Om enige richting te geven volgen hieronder een aantal tips voor de toepassing van CSP. Deze tips zijn afkomstig van <https://internet.nl>.

1. default-src moet gedefinieerd zijn, zodat er een restrictieve default fallback policy bestaat voor door jouw website gebruikte bron-typen waarvoor geen specifieke policy is gedefinieerd. De waarde dient of 'none' of 'self' te zijn. Aanvullend kunnen de domeinnaam zelf, of zijn super-domeinnaam of sub-domeinnaam gedefinieerd worden. Naast deze waarden kan 'report-sample' als waarde zijn ingesteld als je wil dat in de 'violation reports' code-voorbeelden worden opgenomen. Verder staan we https: toe, hoewel je dat niet nodig hebt als je aan de onderstaande regels houdt.
2. frame-src moet worden gebruikt (hetzij per definitie of via de fallback naar child-src en default-src) en moet de waarde 'self', 'none' of een specifieke URL hebben, zodat externe bronnen die in jouw website worden geladen verhinderd wordt om andere bronnen te framen of te embedden met behulp van HTML-elementen zoals <frame> en <iframe>.
3. frame-ancestors moet gedefinieerd zijn en moet de waarde 'self', 'none' of een specifieke URL, zodat andere websites verhinderd wordt om jouw website te framen of te embedden met behulp van HTML-elementen zoals <frame>, <iframe>, <object>, <embed>, of <applet>.
4. unsafe-inline, unsafe-eval en unsafe-hashes dienen *niet* gebruikt te worden, omdat deze XSS-aanvallen mogelijk maken.
5. data: dient *niet* gebruikt te worden in default-src, script-src en object-src, omdat dit XSS-aanvallen mogelijk maakt.
6. http: dient *niet* gebruikt te worden als schema, omdat dit het mogelijk maakt dat bronnen worden gedownload over een onveilige verbinding. Gebruik in plaats daarvan https:.
7. * (wildcard voor het schema- en host-gedeelte van een URL) dient *niet* gebruikt te worden in een CSP-richtlijn, omdat daardoor iedere externe bron is toegestaan.
8. 127.0.0.1 dient *niet* gebruikt te worden in een CSP-richtlijn, omdat dit aanvallen via content injection mogelijk maakt in een gecompromitteerd systeem.

Merk op dat we in het algemeen aanbevelen om domeinnamen in CSP-richtlijnen zo specifiek mogelijk toe te staan, wat tot op zekere hoogte automatisch getest wordt in de subtest van internet.nl voor CSP. Bovendien raden we niet aan om alle domeinnamen onder een TLD (*.nl) of SLD (*.gov.uk) toe te staan. Het wordt ook afgeraden om een ander domein onder een SLD toe te staan in default-src (bijvoorbeeld example1.gov.uk toestaan voor example2.gov.uk).

BIJLAGE B: REFERRER-POLICY

Er zijn instellingen van de Referrer-Policy header die niet bijdragen aan de beveiliging. Om enige richting te geven volgen hieronder een aantal tips voor de toepassing van de Referrer-Policy. Deze tips zijn afkomstig van <https://internet.nl>.

Aanbevolen policy-waarden:

1. Geen gevoelige informatie naar derde-partijen
 - no-referrer
 - same-origin
2. Gevoelige informatie naar derde-partijen alleen via beveiligde verbindingen (HTTPS)
 - strict-origin
 - strict-origin-when-cross-origin

Niet aanbevolen policy-waarden:

1. Gevoelige informatie naar derde-partijen mogelijk via onbeveiligde verbindingen (HTTP)
 - no-referrer-when-downgrade (default policy van browsers)
 - origin-when-cross-origin
 - origin
 - unsafe-url