Browsers en browserissues

Voorbeeldproblemen met oudere browsers

Een voorbeeld van kwaadwillig gebruik van JavaScript in oudere browsers heeft betrekking op de knop Sluiten (de 'X' rechtsboven) in een venster. De gebruiker ziet een pop-upwaarschuwing en wordt gevraagd deze te sluiten. Maar als de gebruiker vervolgens op de sluitende X-knop klikte, activeerde JavaScript schadelijke code. JavaScript kan de X besturen in browservensters, die ook pop-ups bevatten. Door manipulatie van het script dat de knop en de waarschuwing heeft gemaakt, dacht de browser dat hij toestemming had om de code te activeren.

Voorbeeldproblemen met recente browsers

Een ander veelvoorkomend voorbeeld begon met oudere browsers, maar er bestaan nog steeds variaties op. De gebruikers zien een pop-up die beweert: 'Uw browser is geïnfecteerd, klik hier om het te repareren' of een soortgelijk bericht. In werkelijkheid is de computer niet geïnfecteerd – tenminste niet totdat de gebruiker op de link klikt, die een virus start. Dit soort virussen kunnen vaak worden veroorzaakt door elke toetsenbordactie (zoals CTRL + ALT + DELETE), waardoor ze erg gevaarlijk zijn. U moet voorzichtig zijn om de e-mail eenvoudigweg te sluiten en onmiddellijk te verwijderen.

De JavaScript-statuseigenschap kan worden gebruikt om een opgegeven tekenreeks weer te geven in de statusbalk onder aan het browservenster. De statusbalk geeft normaal gesproken het URL-doel weer wanneer u de muisaanwijzer op een link plaatst. De statuseigenschap kan dus worden gebruikt om de URL van een kwaadaardige site te maskeren.

Problemen met 'helperapplicaties' (plug-ins)

Eén van de problemen met de 'Flash' plug-in was dat deze gevoelig was voor misbruik. Vandaar dat deze standaard uitgeschakeld staat en de gebruiker deze – na een waarschuwing – handmatig moet inschakelen.

Als de gebruiker extensies of plug-ins heeft geïnstalleerd voor bepaalde MIME-typen (die bestandsnaamextensies bieden om gegevens voor e-mailoverdracht te helpen coderen en decoderen), kan een hacker JavaScript gebruiken om de applicaties te starten. Deze methode is nogal grof omdat het ook een 'out of memory'-fout zal genereren, die een bewuste gebruiker zal waarschuwen dat er iets mis is. Bovendien, hoewel de applicatie zal worden gestart, is het onwaarschijnlijk dat de lancering zelf schade veroorzaakt zonder hulp van de gebruiker, omdat het antivirusprogramma schadelijke code zal detecteren.

U moet echter begrijpen dat vastberaden hackers alle middelen zullen gebruiken die tot hun beschikking staan. Als ze een van uw programma's kunnen openen (vooral op de achtergrond), kunnen ze de inhoud ervan manipuleren.

Plug-ins (programma's die in de browser worden uitgevoerd, zoals de Flash Player) en helperapplicaties (toepassingen die afzonderlijk van de browser worden gestart, zoals Acrobat voor het lezen van PDF's), vertrouwen op gespecificeerde MIME-informatie om de bestandsindelingen en extensies te definiëren die vereist zijn door verschillende plug-ins of hulpapplicaties. MIME-typen worden gedefinieerd door twee stukjes informatie: het inhoudstype en het subtype. Een geluidsbestand kan bijvoorbeeld een MIME-type audio/x-wav hebben, waarbij 'audio' de categorie of het inhoudstype (een geluidsbestand) identificeert en 'x-wav' het subtype (WAV-bestanden) aangeeft. De 'x' geeft aan dat dit bestandstype geen MIME-standaardtype is. Een ander MIME-type is mogelijk application/msword, wat een Microsoft Word-bestand met de extensie .doc zou aangeven.

Wat gebruikers en ontwikkelaars kunnen doen

Omdat alle browsers – oudere en nieuwere versies – enkele beveiligingsproblemen hebben, moeten alle eindgebruikers ijverig op de hoogte blijven van nieuws van de leveranciers. Als er een beveiligingsprobleem wordt aangekondigd, update dan uw software met de patch of de aanbevolen versie. Onthoud dat een recentere browserversie niet noodzakelijkerwijs veiliger is. U moet uw browser up-to-date houden naar de laatste stabiele versie. Denk er ook aan om de instellingen voor het blokkeren van pop-ups van de browser verstandig te gebruiken en uw antivirussoftware altijd up-to-date te houden.

Als ontwikkelaar kunt u de volgende stap nemen om beveiligingsproblemen in gedachten te houden wanneer u uw pagina's codeert:

- Vermijd praktijken waarvan bekend is dat ze risico's opleveren voor gebruikers.
- Leg de functionaliteit op uw site uit als u denkt dat dit betrekking heeft op uw gebruikers.
- Gebruik zo veel mogelijk op standaarden gebaseerde code; browser-specifieke code minimaliseren.
- Gebruik geen verouderde code.

Script blokkeren

JavaScript kan een grote verscheidenheid aan functionaliteit, interactiviteit en oogverblindende effecten toevoegen aan uw webpagina's. Het levert echter ook enkele veelvoorkomende problemen op. Sommige scripts zijn schadelijk en kunnen de computer van de gebruiker beschadigen. Andere zijn gewoon vervelend. Pop-upvensters kunnen bijvoorbeeld nuttig zijn voor bepaalde taken, maar worden te vaak gebruikt voor doeleinden die gebruikers irriteren, zoals advertenties.

Om deze reden zijn er veel tools van derden beschikbaar waarmee gebruikers JavaScript-code en andere uitvoerbare inhoud kunnen blokkeren. De meeste browsers bieden deze mogelijkheid ingebouwd en in veel gevallen blokkeren ze standaard scripts, tenzij u de instellingen wijzigt. U kunt ook add-ons en plug-ins voor scriptblokkering verkrijgen die zijn ontworpen voor specifieke browsers.

De AdBlock Plus-plug-in: Deze plug-in blokkeert alle vormen van reclame, inclusief fly-ins, slide-ins, pop-ups, pop-unders, spyware- en adware-advertenties, Flash-advertenties, rich media en messenger-advertenties.

In Microsoft Internet Explorer kunt u de volgende instellingen bijwerken om JavaScript in of uit te schakelen: Open Microsoft Internet Explorer> Extra> Internetopties> tabblad Beveiliging> Aangepast niveau> Scriptlets toestaan.

De NoScript-add-on: met deze gratis, open source scriptblokkering voor op Mozilla gebaseerde browsers (bijvoorbeeld Firefox, SeaMonkey) kunnen JavaScript, Java, Flash en andere uitvoerbare inhoud alleen worden uitgevoerd als u de bron accepteert als een vertrouwde website.

Google's Chrome Dev op Windows: De Google Chrome-versie Dev 81.0.4040.5 of latere versie voor Windows biedt een optie om JavaScript, plug-ins, cookies en afbeeldingen selectief te beheren in de Google Chrome-browser. Gebruikers kunnen deze inhoud blokkeren of ervoor kiezen alleen vertrouwde bronnen toe te staan die ze specificeren.

Antivirusbeschermingstoepassingen: de meeste antivirussoftware biedt nu pop-upblokkering, malwarebescherming en waarschuwingen voor onbeveiligde pagina's. Er zijn veel merken beschikbaar, zoals Norton en AVG.

Dergelijke tools zijn bedoeld om te voorkomen dat scripts bekende (en nog niet bekende) beveiligingslekken misbruiken, maar zonder functionaliteit te verliezen door alle uitvoerbare inhoud niet toe te staan.

Overwegingen voor de ontwikkelaar

Scriptblokkering is een verstandige voorzorgsmaatregel voor browsergebruikers, maar hoe beïnvloedt het JavaScript-ontwikkelaars?

U moet zich ervan bewust zijn dat de scripts die u ontwikkelt en opneemt in uw webpagina's kunnen worden gebruikt en genoten door de bezoekers van uw site, of dat ze kunnen worden geblokkeerd. Om deze reden is het verstandig om zorgvuldig na te gaan welke functionaliteit op uw site afhankelijk is van scripts. Moderne webapplicaties zijn echter grotendeels gebaseerd op JavaScript en kunnen in het geheel niet meer functioneren zonder.

Als gebruikers een script nodig hebben om belangrijke activiteiten uit te voeren, moet u gebruikers over het script adviseren zodat ze ervoor kunnen kiezen het toe te staan of om uw site als een betrouwbare bron te accepteren. U kunt ook alternatieve manieren bieden om scriptfunctionaliteit te leveren, zodat gebruikers die uw scripts blokkeren, uw site nog steeds kunnen gebruiken, alhoewel dat tegenwoordig haast niet meer mogelijk is.

JavaScript blokkeren vanuit uw browser

Het kan voorkomen dat u als ontwikkelaar JavaScript volledig moet uitschakelen vanuit uw eigen browser. Een typisch voorbeeld is wanneer u wordt gevraagd een pagina te bekijken waar u niet bekend mee bent, mogelijk vanwege een beveiligingsprobleem. De handigste oplossing is om JavaScript uit te schakelen, de broncode te onderzoeken en uit te voeren, uw taken uit te voeren en vervolgens JavaScript opnieuw in te schakelen wanneer u denkt dat de site veilig is. Normaal gesproken doet u dit in één browser, zodat u niet al uw browsers hoeft te resetten. Hieronder volgen de stappen voor het uitschakelen van JavaScript in Google Chrome 61.x.

Google Chrome: Selecteer Menu | Instellingen | Geavanceerd | Instellingen voor inhoud | JavaScript.

Scroll naar beneden en selecteer de optie Geavanceerd en selecteer vervolgens de knop Instellingen voor inhoud.

Met de schakelknop kan de vereiste instelling worden gewijzigd, JavaScript toestaan of blokkeren op specifieke websites door de knop Uitzonderingen beheren te selecteren en de URL van de website onder het veld Hostnaampatroon toe te voegen. Van daaruit stelt u het Gedrag in op Blokkeren/Toestaan.

Frames

Op moderne websites worden frames niet vaak gebruikt. Ze zijn op grote schaal vervangen door nieuwere technologieën zoals AJAX, die pagina's kunnen veranderen tijdens het vliegen of met een klik op de knop. Daarom is frame-naar-frame URL-wijziging niet het probleem dat het ooit was. Het komt echter nog

steeds voor en inline frames (iframes), die vaker voorkomen, zijn een bepaald doelwit. Daarom moet u op de hoogte zijn van de beveiligingsproblemen als u met JavaScript gaat werken.

Hoe frames werken

Frames zijn een websitetechniek waarbij het browservenster is verdeeld in twee of meer aangrenzende vensters die inhoud van verschillende URL's kunnen weergeven. Frames zorgen ervoor dat een deel van de pagina statisch blijft, terwijl een ander deel dynamisch kan veranderen. U kunt bijvoorbeeld een frameset-pagina maken met links in één venster die, wanneer erop wordt geklikt, nieuwe pagina's opent in het(de) aangrenzende venster(s).
Frame-inhoud komt meestal van binnen de website, maar kan ook van een andere site komen. De mogelijkheid om inhoud uit andere bronnen te delen, maakt frames kwetsbaar, daarom moet de site-ontwikkelaar de inhoud zorgvuldig controleren. Het zonder toestemming gebruiken van inhoud van een andere website is een inbreuk op het auteursrecht. Ook kunnen frames op uw eigen site worden gekaapt door schadelijke code en worden gebruikt om browsersoftware te infecteren.

Cloaking

Wanneer een frame in een frameset een URL omleidt, geeft de browser nog steeds de URL van het frameset-document weer – in plaats van de URL van de doelpagina – in de adresbalk. Deze techniek wordt soms cloaking genoemd en kan voor legitieme doeleinden worden gebruikt (zoals het weergeven van een meer gedenkwaardige URL aan gebruikers). Het kan ook op een kwaadaardige manier worden gebruikt (zoals het verbergen van een 'phishing'-site als onderdeel van een fraude met websitespoofing).

Inline frames

Een frames-pagina kan ook een inline-frame (iframe) bevatten, dat een afzonderlijk bestand zwevend binnen de pagina weergeeft. Deze frames lijken deel uit te maken van de bovenliggende pagina en bieden over het algemeen geen eigen beveiligingsindicator of adresbalk. Gebruikers (en browsers) vertrouwen ze dus impliciet. Dit beveiligingsprobleem wordt urgenter omdat iframes soms worden gebruikt voor wachtwoordinvoer op sites, waar het beschermen van de integriteit van informatie essentieel is. Als uw site een reputatie krijgt voor beveiligingsproblemen met gebruikersinformatie, hebt u veel minder bezoekers.

Een voorbeeld van inline frames is het insluiten van een YouTube-video op een webpagina:

```
<iframe width="560" height="315" src="https://www.youtube.com/embed/ESL52Fqimqo"
frameborder="0" allow="accelerometer; autoplay; encrypted-
media; gyroscope; picture-in-picture" allowfullscreen></iframe>
```

Browserbeperkingen

Historisch gezien kon elk venster de locatie van een ander venster veranderen. Maar dit werd een probleem omdat het betekende dat het gebruik van een frame of iframe-venster voor inloggen onveilig was (een kwaadwillende site kan uw inlogpagina vervangen door een vervalste versie). Het stelde hackers ook in staat om uw frames-pagina te kapen door een link om te leiden om ongewenste inhoud in een frame weer te geven.

Vanwege deze problemen zijn er in de loop van de tijd beperkingen opgelegd die pogen locatiewijzigingen in browservensters te beperken, vooral bij het gebruik van elk type frame. En zoals u weet, kunt u JavaScript gebruiken om URL's om te leiden naar het locatieobject.

Er zijn nu regels die het soort pagina's beperken waarnaar u in een frame of iframe kunt omleiden. Als u op een frameset-pagina probeert de URL van een frame onjuist te wijzigen vanuit een ander frame met

behulp van JavaScript, retourneren veel moderne browsers een foutmelding die zegt: 'Onveilige JavaScript probeert toegang te krijgen tot frame met URL.' En de URL verandert niet. Er zijn manieren om dit te omzeilen, maar dergelijke procedures worden niet als beste praktijk beschouwd en de code is vrij complex.