مروری بر حملات CSRF

نویسنده: پویا دانشمند

whh_iran@yahoo.com

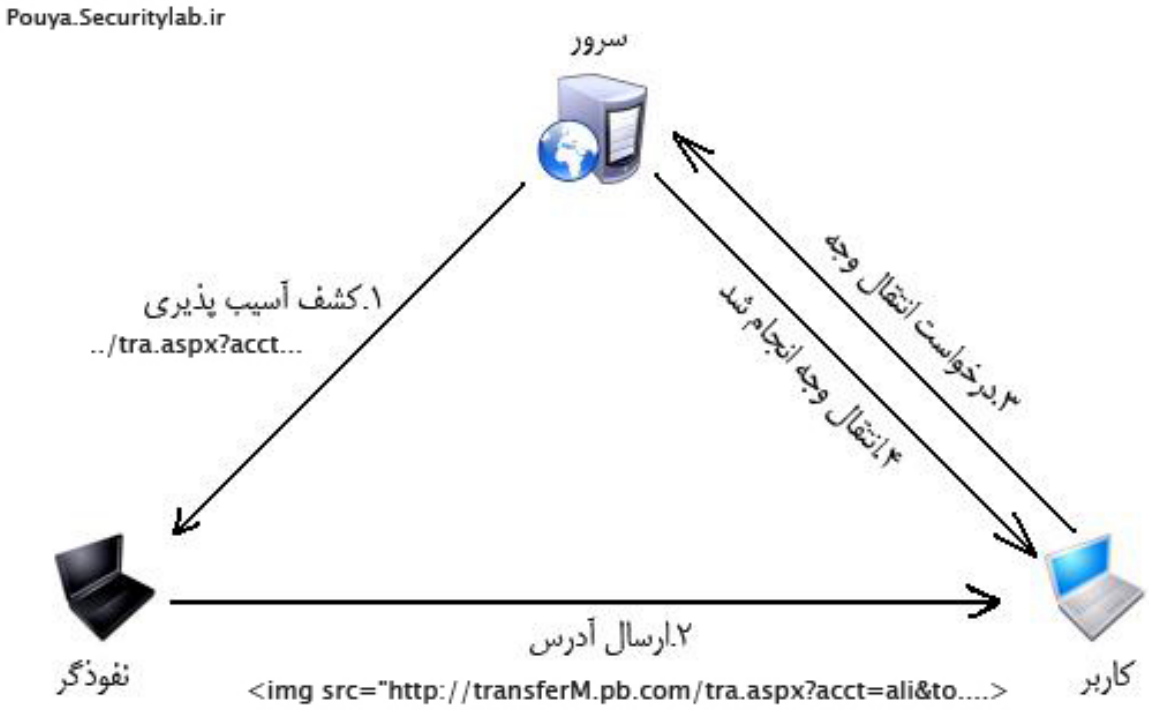
http://pouya.securitylab.ir

نگارش اول ، اسفند 88

مقدمه

کاربران در وب سایت ها معمولا با استفاده از چند کلیک به انجام امور خود میپردازند ، این اعمال اغلب با استفاده از کلیک کاربر بر روی لینک و ارسال فرم و .. انجام و پردازش می شود در حملات Cross Site Request Forgery که به طور اختصار CSRF خوانده می شود نفوذگر از اعتماد وب سایت به کاربر استفاده کرده و به طور کلی تمام حمله از طرف یکی از کاربران سایت شکل میگیرد ، شناسایی این گونه حملات اغلب سخت و طاقت فرساست همچنین میتوان به این آسیب پذیری به عنوان یک آسیب پذیری قدیمی و همین طور پر استفاده اشاره کرد که اولین نشانه های آن در سال 2000 دیده شده و در سال 2008 اطلاعات بسیاری از کاربران سایت معروف eBay با استفاده از همین روش مورد سرقت قرار گرفت و در اوایل همان سال مشتریان یکی از بانک های مکزیک قربانیان این حمله شدند .

حمله CSRF در یک نگاه



نحوه شکل گیری حملات CSRF

حملات CSRF مرورگر قربانی را وادار به انجام و ارسال دستورات مورد نظر خود میکند ، البته این مشکل به علت برنامه نویسی غلط و همین طور استفاده از بعضی توابع شکل میگیرد با هم مثال زیر را مرور و تشریح می کنیم :

علی برای صرفه جویی در زمان انتقال وجه را از طریق اینترنت و سایت بانک انجام میدهد ، علی قصد حواله مبلغ ده هزار تومان را به حساب مجید دارد ، فرآیند انتقال وجه به شکل زیر است :

```
POST http://transferM.pb.com/tra.aspx HTTP/1.1

…

Content-Length: 20;

acct=majid&tot=10000
```

در واقع انتقال به این شکل انجام می شود :

```
http://transferM.pb.com/tra.aspx?acct=majid&tot=10000
```

پس این حملات بر پایه HTTP request اجرا و اعمال میشود.

علی با دیدن این فرآیند تصمیم به استفاده از این مشکل میگیرد و یک URL با مشخصات زیر آماده میکند :

```
http://transferM.pb.com /tra.aspx?acct=ali&tot=100000
```

با اجرا کردن این URL فوق مبلغ مورد نظر برای ali ارسال می شود بدیهیست که هر شخصی در ضمن وارد بودن به حساب کاربری خود این عملیات را شکل می دهد و این سیر فقط در صورت نداشتن موجودی و یا وارد نبودن کاربر به حساب بانکی خود با شکست روبرو خواهد شد .

در ادامه نفوذگر باید به شکلی آدرس بالا را در مرورگر طرف مقابل اجرا کند ، ولی قبل از آن باید آدرس مورد نظر را به شکل مبهمی برای کاربر ارسال کرد زیرا مبتدی ترین کاربران هم به آدرس بالا شک خواهند کرد ، پس با یکی از اشکال زیر آدرس را مبهم ساخته و پس از آن آدرس را برای شخص مذکور می فرستیم .

تبدیل آدرس به فرمت hex (مبهم سازی) :

```
http://transferM.pb.com/tra.aspx%3Facct%3Dali%26tot%3D100000
```

در برخی موارد کاربر به علت وجود علائم و شلوغی URL به آن توجه کافی نمی کند.

در قالب یک تصویر:

```
<"img src="http://transferM.pb.com/tra.aspx?acct=ali&tot=100000" width="2" height="2>
```

کد بالا در صفحه مجید نمایش داده نخواهد شد و مرورگر این تصویر را رندر نمی کند ، این در حالیست که مجید به طور غیر مستقیم روی لینک کلیک کرده و انتقال وجه انجام شده است ، این تصویر ممکن است در سورس صفحه یک وبلاگ یا نامه الکترونیکی و یا هر جای دیگر گنجانده شده باشد.

به صورت IFRAME یا Script :

```
<iframe src=" http://transferM.pb.com/tra.aspx?acct=ali&tot=100000">
```

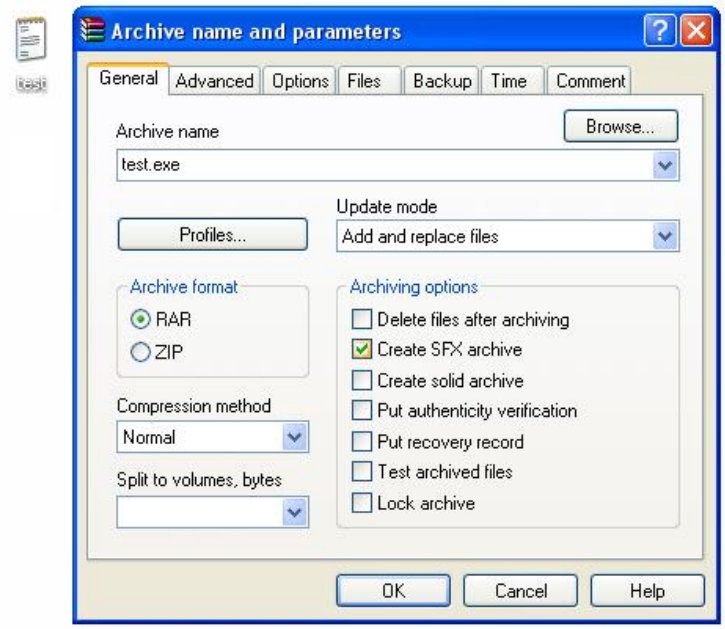
```
<script src=" http://transferM.pb.com/tra.aspx?acct=ali&tot=100000">
```

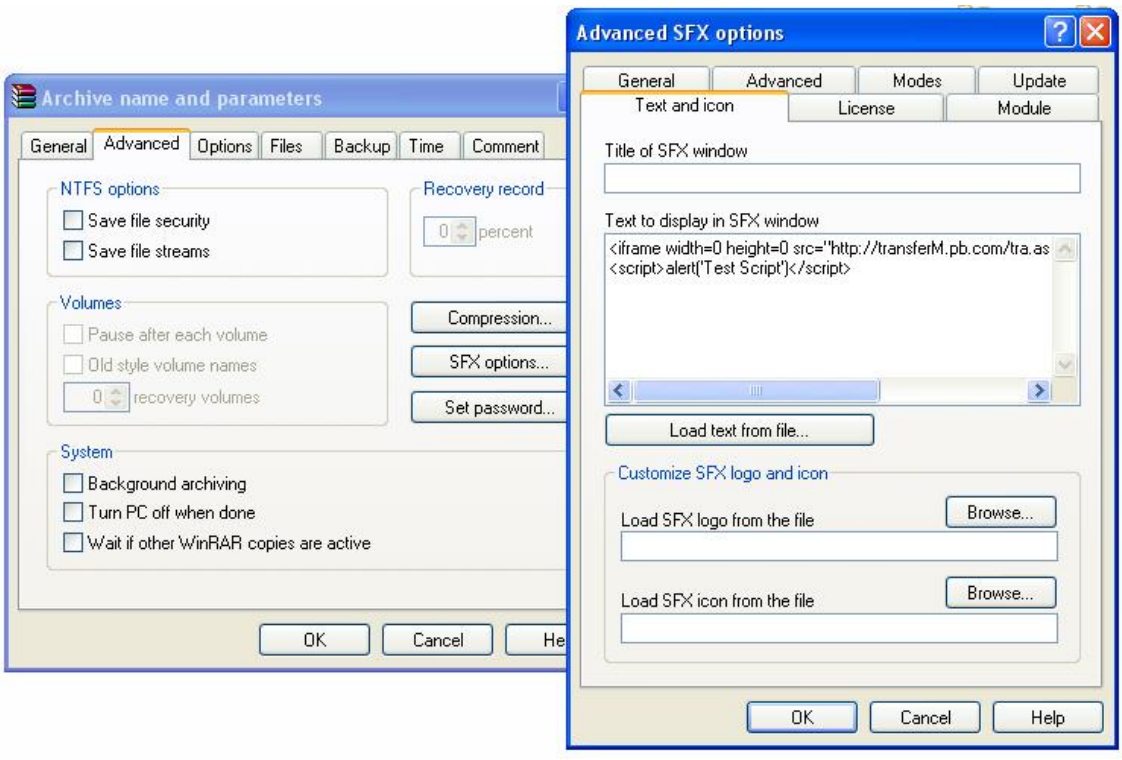
یا حتی به صورت یک فایل فلش :

```
Import flash.net.URLRequest;
Import flash .system.Security;
var url = new URLRequest("http:// transgerM.pb.com/…");
 var sec = new URLVariables();
url.method = "POST";
url.data = sec;
sendToURL(url);
stop();
```

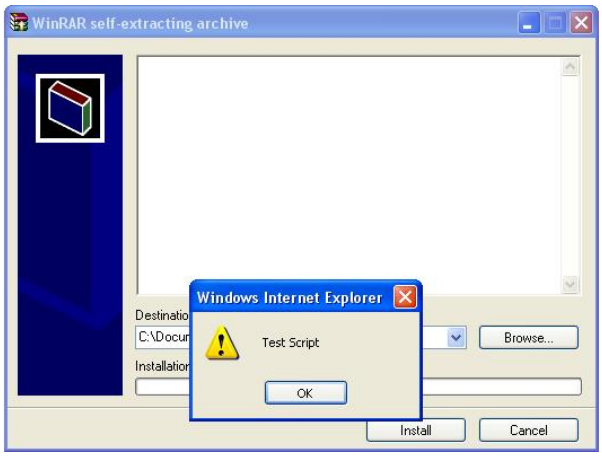
توجه داشته باشید که در بعضی از موارد بالا لازمه ی حمله باز کردن یک صفحه وب سایت نیست مثلا" در مورد Iframe نفوذگر با اندکی خلاقیت توانایی آن را خواهد داشت که کد مورد نظر را در یک فایل اجرایی قرار بدهدکه ساده ترین نمونه آن استفاده از قابلیت های نرم افزار winrar است :

در ابتدا فایلی را برای ساخت آرشیو انتخاب می کنیم ، بعد از آن گزینه SFX Archive را فعال کرده سپس در قسمت SFX Option روی تب Text and icon کلیک کرده و در پنجره آدرس را به شکل iframe قرار می دهیم .





یک پیغام هم برای آزمایش و صحت اجرایی فایل به آن اضافه می کنم و بعد از ساخت و اجرا مشاهده خواهیم کرد که همه چیز بدرستی انجام می شود .



و در نهایت آدرس مورد نظر ما در ماشین کاربر اجرا خواهد شد .

صفحه مورد نظر به خاطر تنظیم طول و عرض صفر ، توسط کاربر قابل مشاهده نخواهد بود ، به همین ترتیب می-توان انواع و اقسام فایل ها را برای این نوع حملات آماده کرد.

حفاظت در برابر حملات CSRF

پیشنهاداتی برای برنامه نویسان :

- تست عمومی برای جداسازی ماشین از انسان (Captcha)
- بررسی و منقضی کردن کوکی (Cookie)
- استفاده از شناسه معتبر (Valid Token)
- از متود GET استفاده نکنید ، به طور مثال در PHP :

به جای استفاده از $_REQUEST['var'] از $_POST['var'] استفاده کنید.

یا در ASP.Net :

Request.Params["var"]; را با Request.Form عوض کنید .

پیشنهاداتی برای کاربران :

- بعد از اتمام کار در سایت حتما بر روی خروج (log off) کلیک کنید تا به طور کامل از سایت خارج شوید.
- در صورتی که در حال مشاهده سایت خاصی هستید از مرور کردن سایت های دیگر بپرهیزید.
- لینک ها و فایل های مشکوک را بطور کلی باز نکنید .