Implementation of Graphic User Interface

The ESEC lecture presented by Mr. Eric Khoury talked about how his team designed a software program that can model and analyze basketball games for the Raptors team. The software's graphic interface was programmed using Java Application due to his team's overall proficiency of Java. He briefly mentioned that Java Applets are also a possibility. However, he did not mention the reason as to why Java Application is chosen over Java Applets. This extension will be exploring the details about Java Applets to summarize potential reasons as to why Mr. Khoury's team chose one over the other.

Java Applications are standalone executables where one can download and setup a package of files. Once the initial installation completes, internet connection is no longer needed and the application can freely run having access to the downloaded files. Whereas Java Applets are designed for web browsers and usually imbedded in the webpage.

The disadvantage of Java Applications is the number of installations required to keep an application up to date. In the presence of new data or files, one must commit the effort in downloading the changes again, called applying patches. For example, for every game played by the Raptors, a recording of the game is stored, however players will not have access to this new game unless they update their application by downloading the latest game file. With a lot of games being constantly played, the procedure of applying patches will become tedious.

Java Applets can solve the issues associated with Java Applications. Applets are linked to web browsers that are constantly updated. In the presence of a new basketball game, as long as the recording of the new game is uploaded to the browser, the users will have instant access to the new file. However, Java Applets have its disadvantages as well. Java Applets have very big security issues (Appendix [2]) and risks associated with it, mainly in four categories:

1. Denial of Service Attacks
2. System Modification
3. Invasion Privacy
4. Antagonism

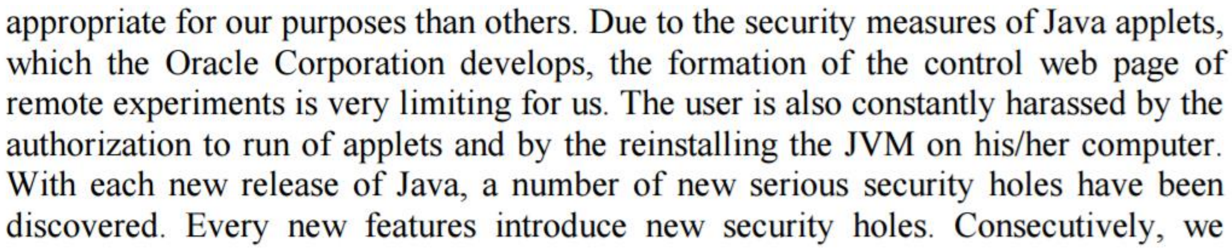
Due to these security issues, it is extremely difficult to implement a Java Applet program. Oracle, the corporation that owns Java, has very strict requirements when it comes to allowing a Java Applet program to be embedded in a browser. The authentication procedure takes a long time and is very tedious for the developers (Appendix [1]). Even when the applet is approved, major web browsers like Google Chrome simply doesn't support any applets at all. The user will have to manually allow the specific applet to be displayed as well as downloading appropriate packages from Oracle to actually view the applet (Appendix [3]).

In conclusion, both Java Application and Java Applet have their disadvantages. However, the inconvenience associated with Java Applet's security issue simply makes it not worth the time for Mr. Khoury's team. The easier way out is to choose Java Application, where it's disadvantage does not require extra work from the developers, which could be a reason for why Mr. Khoury's team chose this option.

Appendix

1.

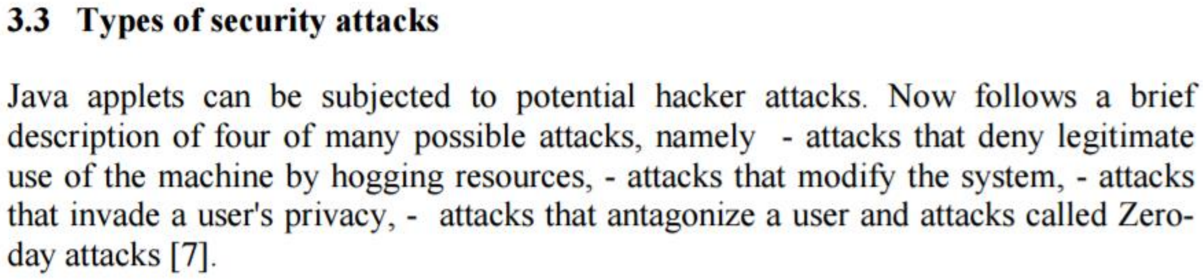
Petra Špiláková, Roman Jašek and František Schauer, “Security Risks of Java Applets in Remote Experimentation and Available Alternatives”, Tomas Bata University in Zlín, Faculty of Applied Informatics, Nad Stráněmi 4511, 760 05 Zlín, Czech Republic



appropriate for our purposes than others. Due to the security measures of Java applets, which the Oracle Corporation develops, the formation of the control web page of remote experiments is very limiting for us. The user is also constantly harassed by the authorization to run of applets and by the reinstalling the JVM on his/her computer. With each new release of Java, a number of new serious security holes have been discovered. Every new features introduce new security holes. Consecutively, we

2.

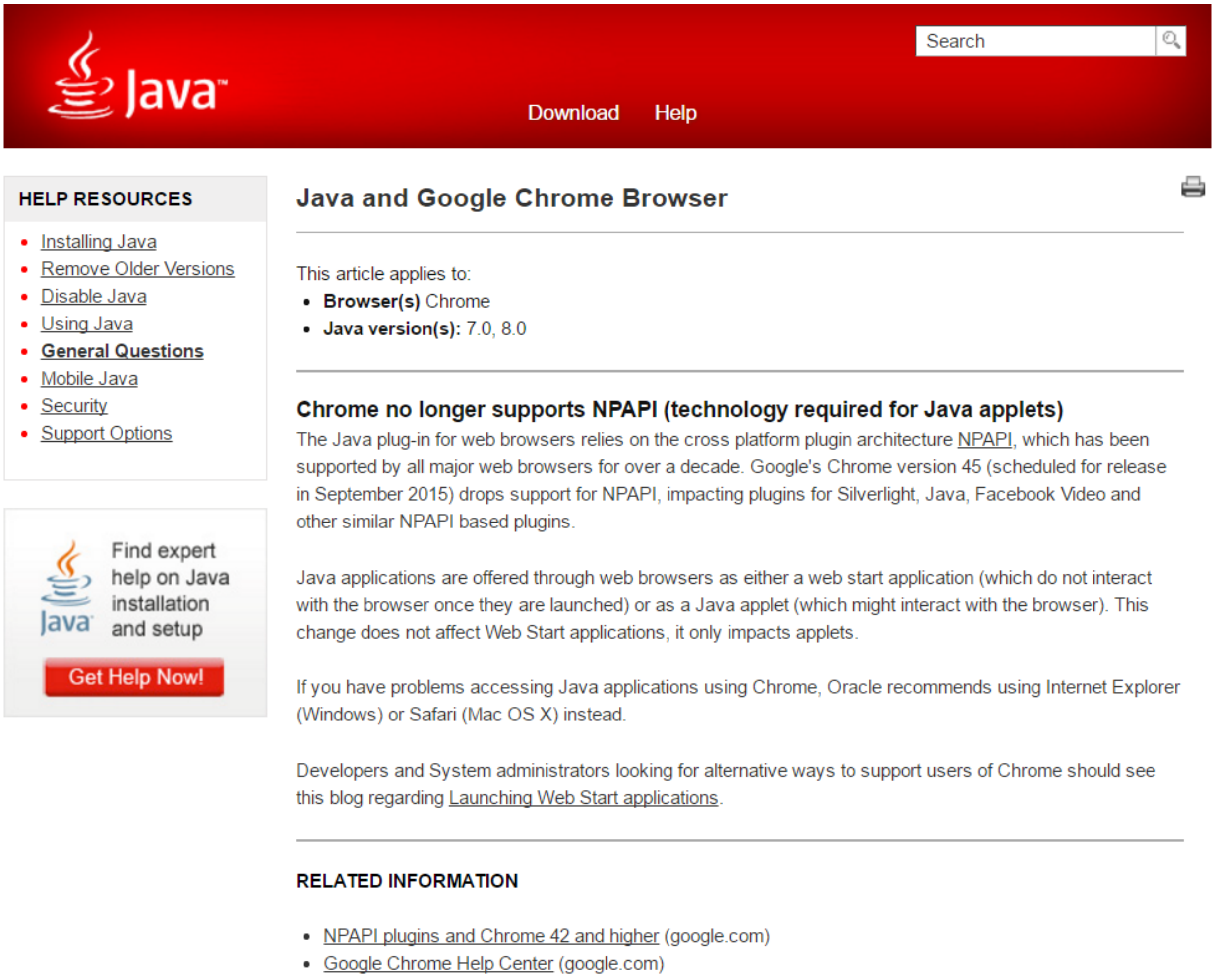
Petra Špiláková, Roman Jašek and František Schauer, “Security Risks of Java Applets in Remote Experimentation and Available Alternatives”, Tomas Bata University in Zlín, Faculty of Applied Informatics, Nad Stráněmi 4511, 760 05 Zlín, Czech Republic



3.3 Types of security attacks

Java applets can be subjected to potential hacker attacks. Now follows a brief description of four of many possible attacks, namely - attacks that deny legitimate use of the machine by hogging resources, - attacks that modify the system, - attacks that invade a user's privacy, - attacks that antagonize a user and attacks called Zero-day attacks [7].

3. Major browsers don’t support applets



Java and Google Chrome Browser

This article applies to:

- **Browser(s)** Chrome
- **Java version(s):** 7.0, 8.0

Chrome no longer supports NPAPI (technology required for Java applets)

The Java plug-in for web browsers relies on the cross platform plugin architecture NPAPI, which has been supported by all major web browsers for over a decade. Google's Chrome version 45 (scheduled for release in September 2015) drops support for NPAPI, impacting plugins for Silverlight, Java, Facebook Video and other similar NPAPI based plugins.

Java applications are offered through web browsers as either a web start application (which do not interact with the browser once they are launched) or as a Java applet (which might interact with the browser). This change does not affect Web Start applications, it only impacts applets.

If you have problems accessing Java applications using Chrome, Oracle recommends using Internet Explorer (Windows) or Safari (Mac OS X) instead.

Developers and System administrators looking for alternative ways to support users of Chrome should see this blog regarding Launching Web Start applications.

RELATED INFORMATION

- NPAPI plugins and Chrome 42 and higher (google.com)
- Google Chrome Help Center (google.com)