SSI/CGI Standard Practice for www.uni.edu

Technology Planning Guideline or Document

Introduction

Recently, there has been an increased need for enhanced server side functionality on the main UNI UNIX web server. Due to the latest UNI web site redesign and other projects, it is now necessary to revisit the university procedures regarding Server Side Includes (SSI) and Common Gateway Interface (CGI) scripts. SSI enables code to be run on a server whose output is then used to dynamically generate data for a webpage. The code for SSI is embedded within an HTML file, so the file must be parsed by the web server. Since this has an impact on performance, SSI files are usually denoted with a .shtml extension to prevent the server from having to parse all html files.

CGI scripts are executable code that produce HTML or other web ready data. They can be written in a number of languages, most commonly Perl and C. There are a vast number of CGI scripts available on the Web to support webboards, guestbooks and other functionality.

Due to security issues which will be explained in this document, the use of these technologies must be controlled.

Current Standard Practice

The current procedure for user supplied CGI scripts is completely open. That is, any user may upload an executable CGI script with the extension .cgi and execute permissions to their webspace on www.uni.edu [1]. This makes creating dynamic websites easier for the end user, as the CGI functionality is turned on by default. The current SSI procedure is more restrictive. If a user can justify the need for SSI functionality in their webspace, it will be turned on for them by ITS staff. This allows for administrative control over what users may do with SSI technology.

The Problem

With the current standard practice it is possible that both SSI and CGI functionality could be enabled for the same directory. This could be a major security problem if a poorly written CGI script allows for unchecked HTML code to be generated from user input. This could result in SSI exec code being embedded in a guestbook, for instance, giving a web visitor the ability to execute code on the server. This could be used to gain inappropriate access to the web server.

Proposed Standard Practice

Due to the above security concerns, a new standard practice for SSI and CGI is needed. ITS will work with the maintainers of current sites to address any potential conflicts with the new standard practice. For all accounts, SSI/CGI functionality will be enabled in a limited and controlled manner. Specifically, SSI exec functionality (the most problematic aspect of SSI) will be disabled by default. All other SSI functionality will be enabled for the entire user account, except for all cgi-bin directories, where no SSI functionality will exist. CGI access will be enabled only in cgi-bin directories. SSI functionality will never be enabled for these directories to prevent the risk detailed in the previous section. If additional functionality is needed, it will be added on a case by case basis at the discretion of ITS staff. Periodic monitoring of the use of this technology will help to identify potential problems. For example, say a new user has a single directory in their home directory called project1 and they need both CGI and SSI access for their pages. Their home directory would be setup as

follows:

Directory	SSI	CGI
`/export/home/username/`	yes	no
`/export/home/username/project1`	yes	no
`/export/home/username/cgi-bin`	no	yes

Implementation

The new SSI/CGI procedure will be implemented as follows: Documentation of the new procedure and how it affects campus web developers will be written and approved by ITS and made available to the campus web development community. A web based form will be posted on the web allowing web developers a channel to request special SSI/CGI privileges. This form will provide information that may be used by ITS to review the proposed code. On a predefined date, the current procedure will cease, and the new procedure will be put in place. Any existing accounts that may be modified without breaking current web functionality will be modified to comply with the new procedure. From this date on, all new accounts will be created under the new procedure.

Conclusion

By making some simple changes to the current SSI/CGI standard practice, the system security and functionality can be improved on www.uni.edu [1] resulting in a more secure campus webspace for ITS staff, users and visitors.

October 5, 2000

Source URL: http://www.uni.edu/its/policies/ssicgi-standard-practice-wwwuniedu

Links:
[1] http://www.uni.edu