Pulse2011

Cleaning your TAM WebSEAL with AJAX and other Web 2.0 Challenges.

- Presenter's Name David Siviour
- Presenter's Title Systems Designer
  CenITex

Agenda

- What you should gain from this talk.
- Introduction
  - Briefing on where WebSEAL sits in the TAM eBusiness offering.
  - What is a reverse proxy.
  - How WebSEAL basically works.
  - How we have deployed WebSEAL.
- Web 2.0, AJAX and WebSEAL.
  - How does Web 2.0 and AJAX give WebSEAL heartburn.
- What solutions does WebSEAL have that allow it to act as an antacid.
  - Problems that each solution does not cover.
- Case Study of the implementation of Kronos.
  - What were the issues.
  - What solutions we tried.
  - What was the final solution.
  - How the final solution was implemented.
- How has this changed the way we handle these issues now.
- Questions and Feedback

What you should gain from this talk.

- What is WebSEAL?
- What are WebSEAL/Reverse Proxies meant to do?
- What problems do Reverse Proxies have with Applications?
- What basic methods does WebSEAL have to avoid these problems?
- What problems does Web2.0 and AJAX bring that were not readily seen before?
- What tools can you use to assist in fixing these problems.

Introduction

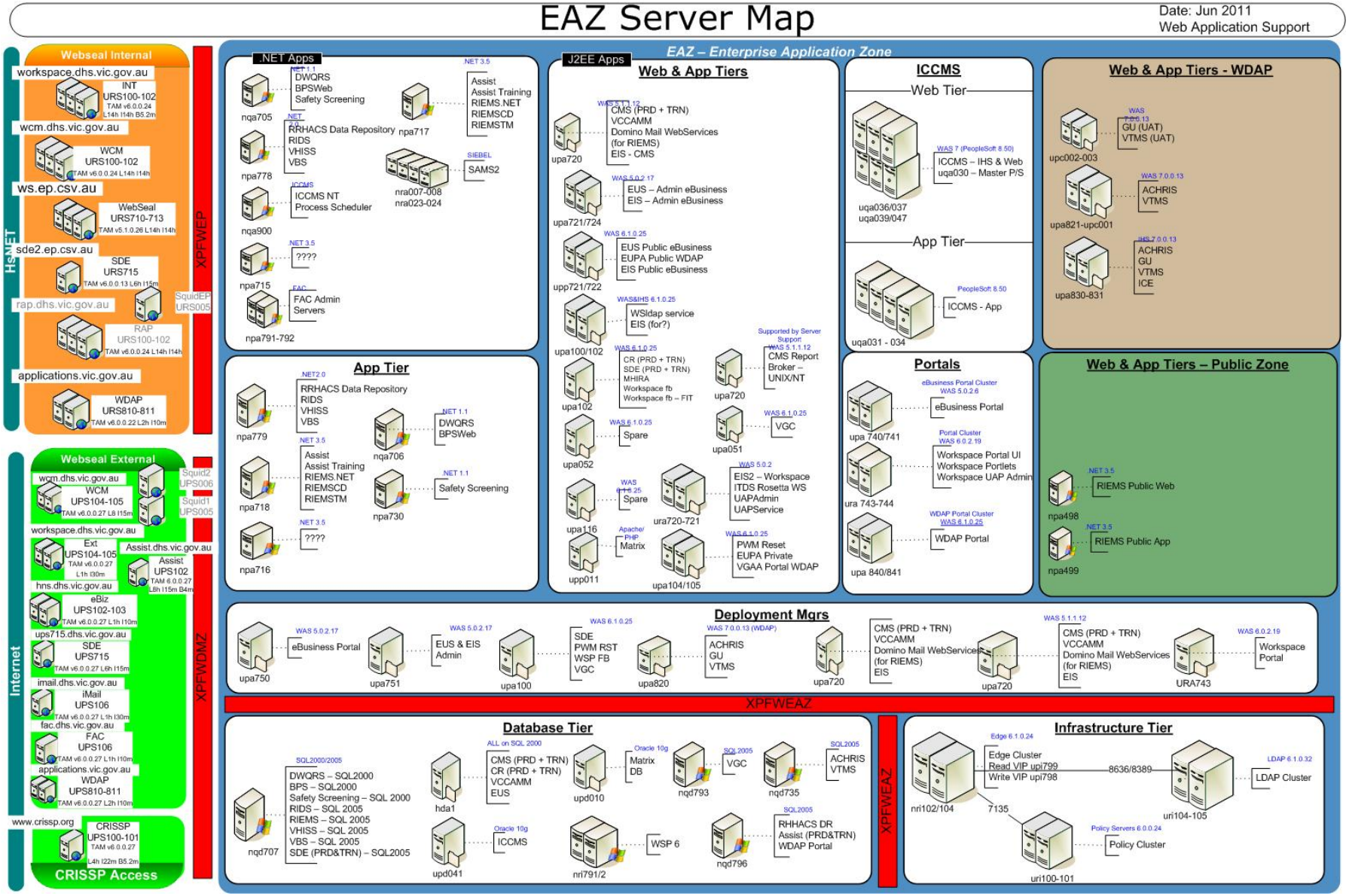
- Previous papers
- This is from an Administrator's Point of View.
- CoTS Applications and applications with a lack of source code.
- Where does WebSEAL sit in the TAM eBusiness offering.
- What is a reverse proxy.

"a **reverse proxy** is a type of proxy server that retrieves resources on behalf of a client from one or more servers." (Wikipedia)

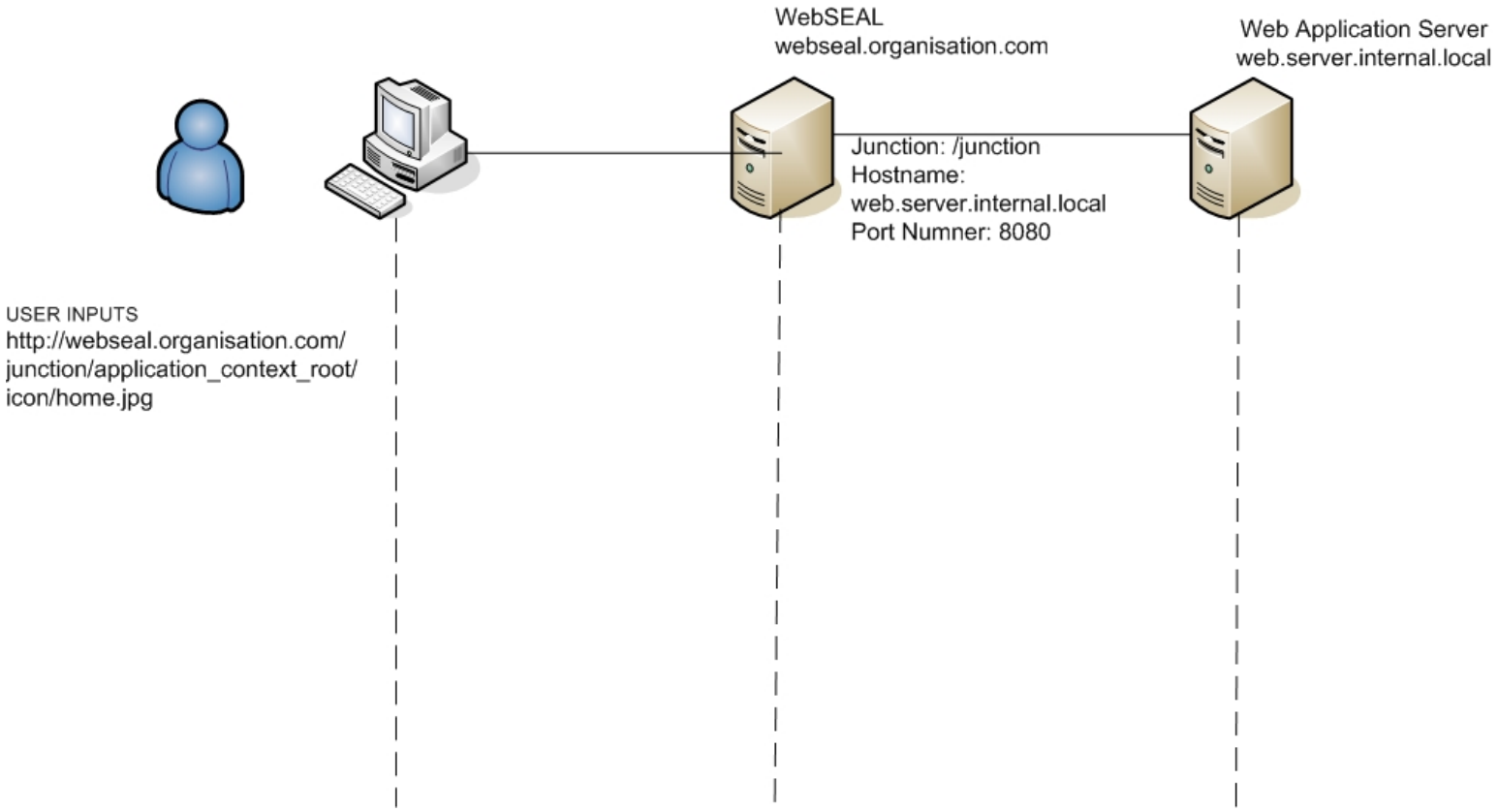
Advantages of a Reverse Proxy

- Obfuscate.
- Load balance.
- Central SSL endpoint.
- Certified Authentication and Authorisation engine.
- Ease SSO deployments.
- Multifactor authentication capabilities.
- Application security capabilities.
- Inbuilt DoS Defence.

WebSEAL/Reverse Proxy Deployment

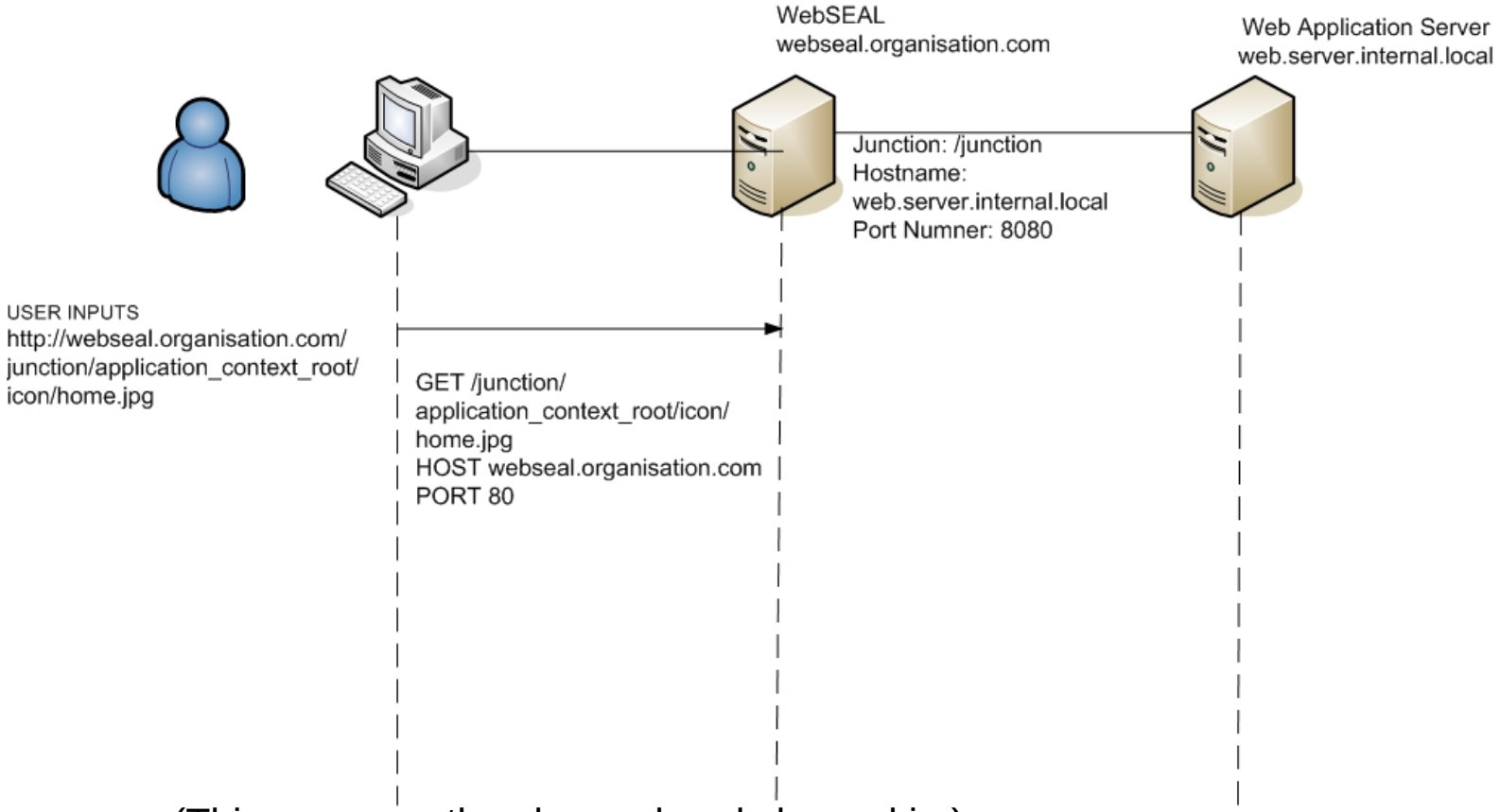


How WebSEAL basically works.



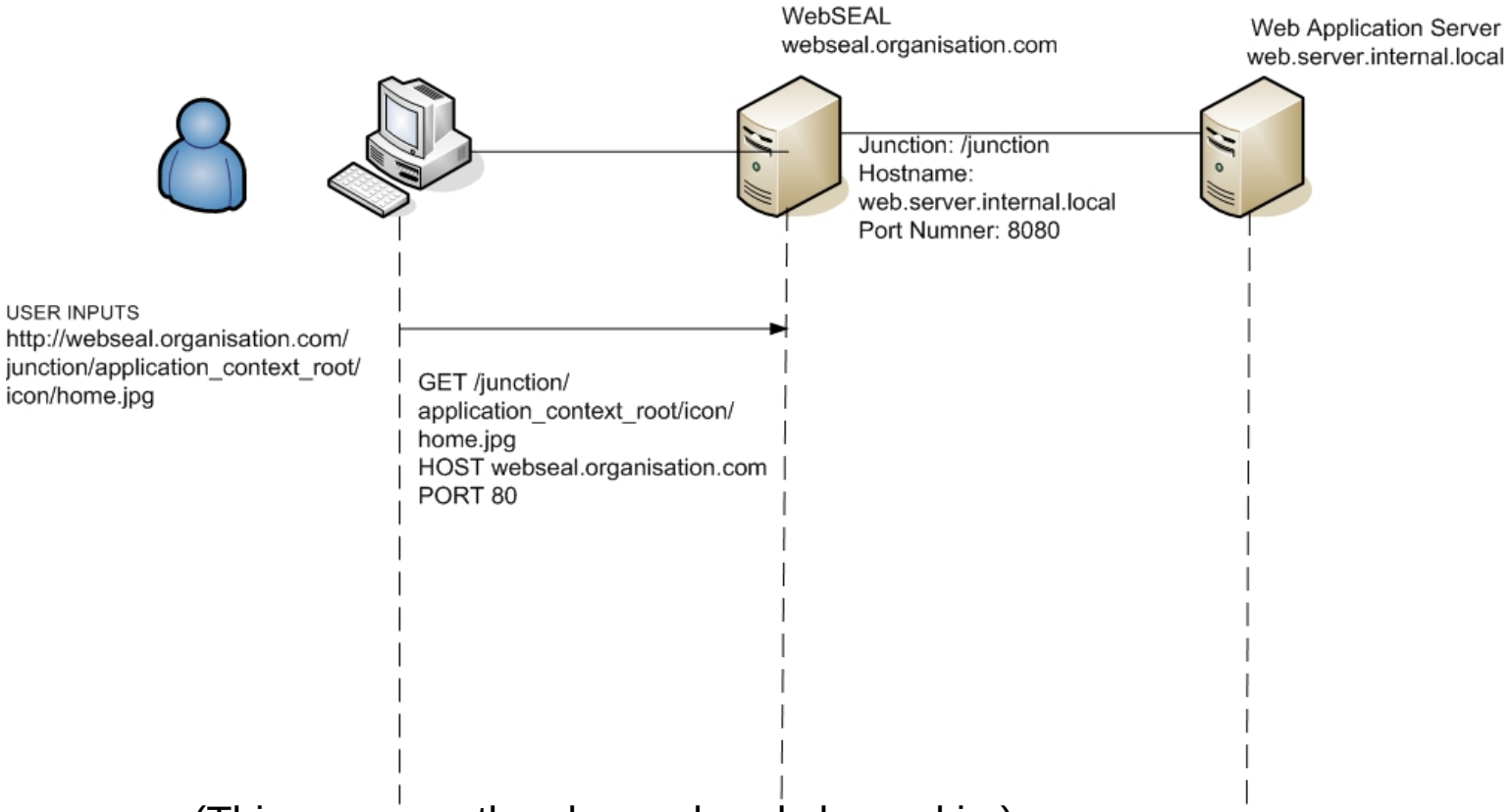
- e.g. (This assumes they have already logged in.)
- http://webseal.organisation.com/junction/application_context_root/icon/home.jpg
  - GET /junction/application_context_root/icon/home.jpg
  - HOST webseal.organisation.com
  - PORT 80

How WebSEAL basically works.



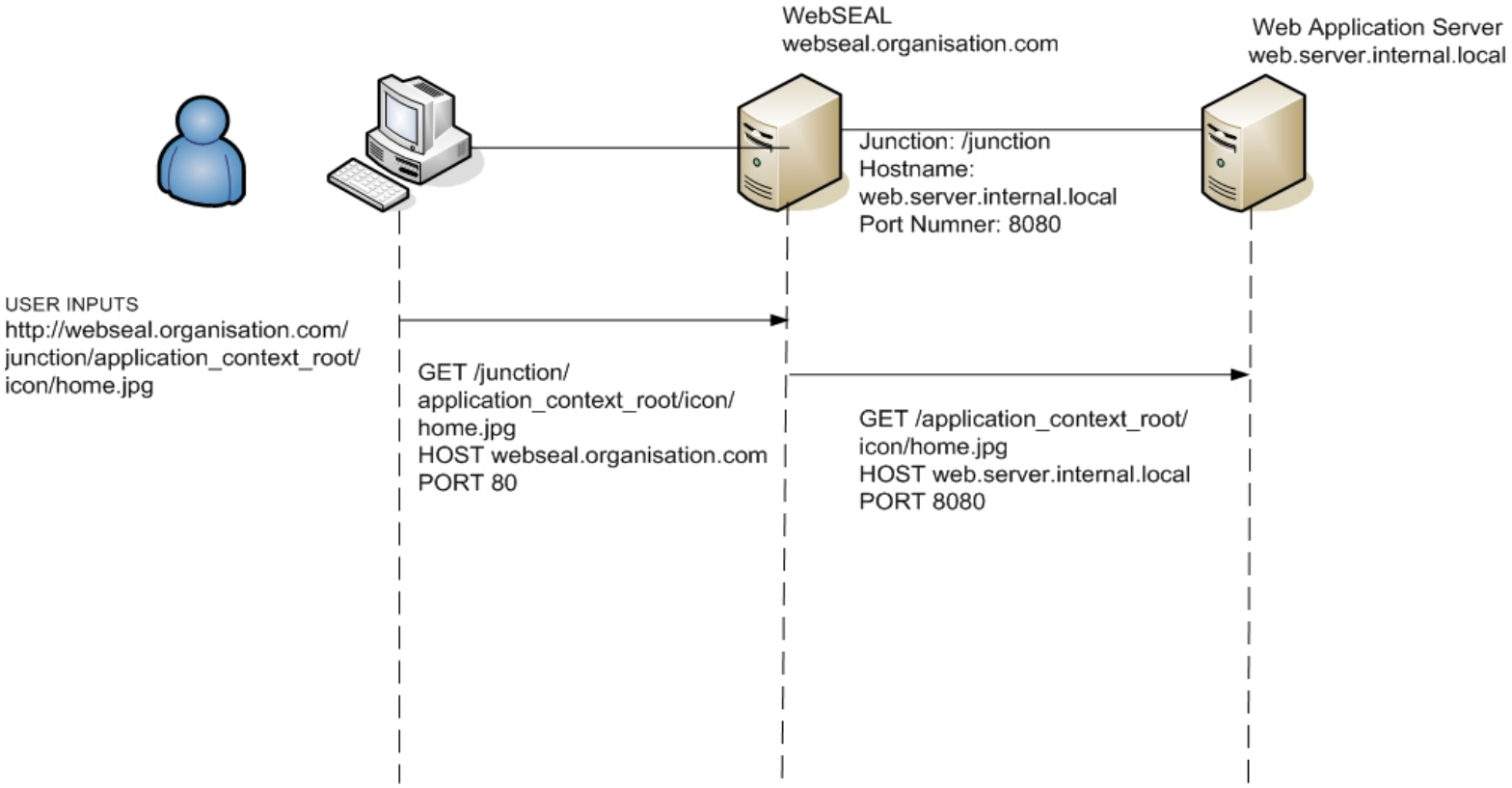
- e.g. (This assumes they have already logged in.)
- http://webseal.organisation.com/junction/application_context_root/icon/home.jpg
  - GET /junction/application_context_root/icon/home.jpg
  - HOST webseal.organisation.com
  - PORT 80

How WebSEAL basically works.



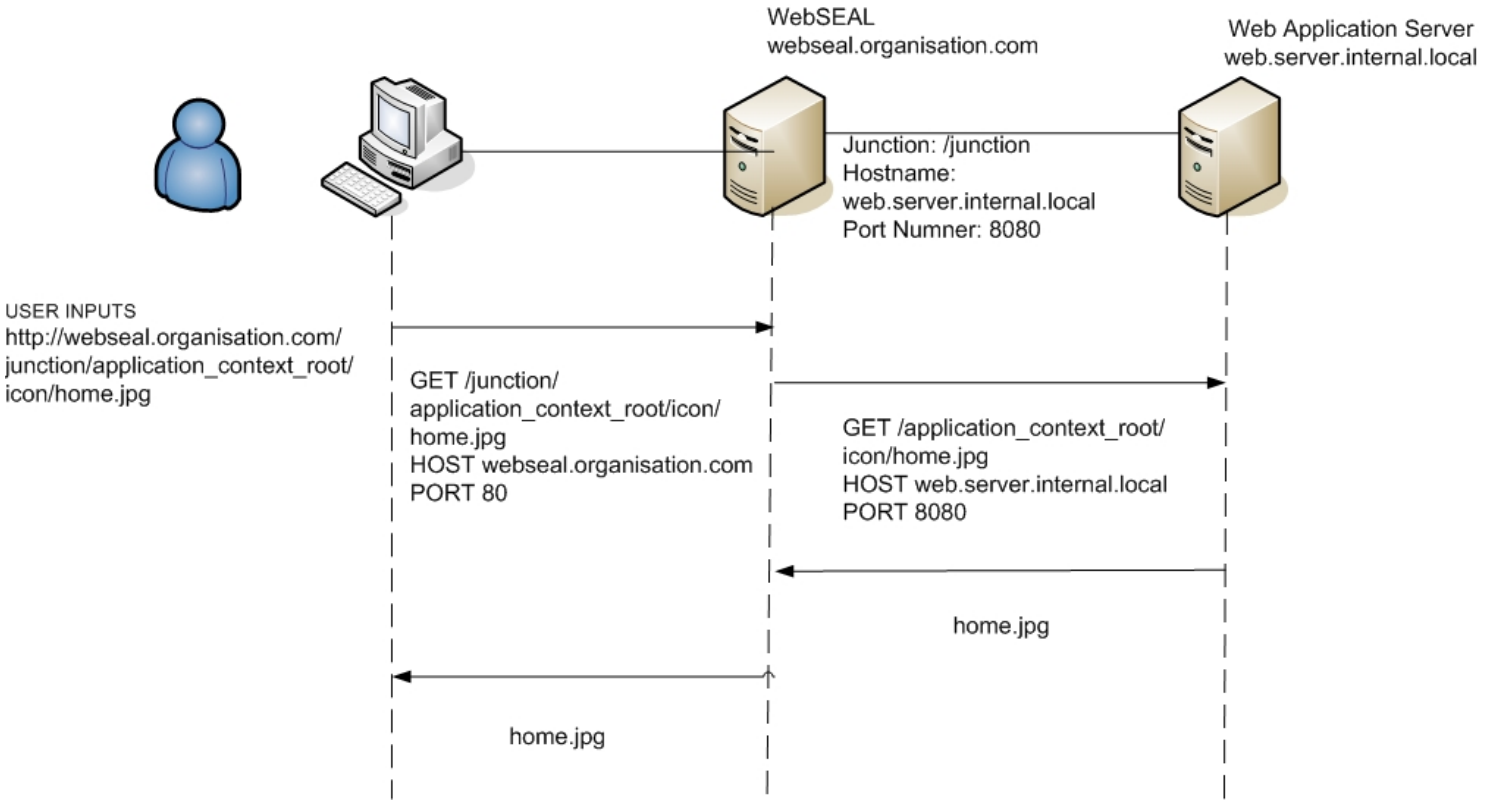
- e.g. (This assumes they have already logged in.)
- Junction includes:
  - Host name: web.server.internal.local
  - Port number: 8080
  - Junction name /junction

How WebSEAL basically works.



- e.g. (This assumes they have already logged in.)
- To the Web Server
  - GET /application_context_root/icon/home.ico
  - HOST web.server.internal.local
  - PORT 8080

How WebSEAL basically works.



- e.g. (This assumes they have already logged in.)
- To the Web Server
  - GET /application_context_root/icon/home.ico
  - HOST web.server.internal.local
  - PORT 8080

Basic Issues and Solutions WebSEALs have with Applications.

- Mapping URLs being returned to the Client.
  - Relative URLs e.g. ../../images/appimage.jpg
    - Natively handled
  - Server Relative URLs e.g. /images/appimage.jpg
    - JMT (Junction Mapping Table)

      e.g.

      /junction /images/*

      this maps all /images/ requests to the /junction server
    - -j (-J tailer –J inhead)
  - Absolute URLs http://web.server.internal.local:8080/images/appimage.jpg
    - script-filtering=yes with filter-nonhtml-as-xhtml=no

Web 2.0, AJAX and WebSEAL

How does Web 2.0 and AJAX give WebSEAL heartburn?

- Javascript filelets: This is the use of small files of javascript being loaded within <script> tags.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
    …
    <script type="text/javascript" src="/wfc1static/applications/suitenav/scripts/TakeMeTo.js"></script>
    …
```

Where the contents of TakeMeTo.js start like this.

```
/**
  * depends on suite-nav-menu.js, dom-utils.js
  */
function TakeMeTo(controlId, dropdownId)
{
    this.element = document.getElementById(controlId)
…
```

- XML with broken URLs:
- XML and other MIME types:
- AJAX Frameworks:
- WebSEAL Session timeout:

WebSEAL Solutions

What solutions does WebSEAL have that allow it to counteract these issues.

- Filtering extra MIME types
- Transparent junctions
- Virtual Host junctions

Things to watch for or avoid in WebSEAL/junction configuration

- -j on a junction configuration
- filter-nonhtml-as-xhtml=yes

Case Study: Implementation of Kronos.

- Kronos is a CoTS Solution for Human Resource Management. In this particular case Rostering and Attendance.
- WebSEAL was to handle Desktop Single Signon through to the application.

What were the issues.

- Server Relative URLs:
- Invalid MIME type:
- Host Header:

What solutions we tried.

- Basic WebSEAL junction since we had a mixed 5.1 6.0 WebSEAL environment.
- JMT for the Server Relative URLs
- -v websealhostname.com
- Extra MIME types in webseald.conf

What was the final solution.

- Virtual host junction:
- IIS compatible rewrite mod:
- Application configuration:

How has this changed the way we handle these issues now.

- Transparent Path Junctions
- Virtual host Junctions
- BUT NO Silver bullets

Resources

- **AJAX techniques within a Tivoli Access Manager WebSEAL Environment**
  Considerations, issues, potential solutions and best practices
  Peter Tuton (ptuton@au1.ibm.com), Software Engineer, IBM
  Grant Murphy (gmurphy@au.ibm.com), Software Engineer, IBM Australia
  http://www.ibm.com/developerworks/tivoli/library/t-ajaxtam/index.html
- **Ajax in a network: Security and topology challenges of aggregating content from multiple sites in an Ajax architecture**
  Using WebSphere Application Server Feature Pack for Web 2.0 and Tivoli Access Manager WebSEAL
  Kevin Haverlock (kbh@us.ibm.com), Software Development, IBM
  Peter Tuton (ptuton@au1.ibm.com), Software Engineer, IBM
  Grant Murphy (gmurphy@au.ibm.com), Software Engineer, IBM Australia
  http://www.ibm.com/developerworks/websphere/techjournal/0909_haverlock/0909_haverlock.html
- Http Watch http://www.httpwatch.com/ for both IE and Firefox
- HttpFox https://addons.mozilla.org/en-US/firefox/addon/httpfox/ for Firefox
- Tivoli Access Manager for eBusiness Infocenter: http://publib.boulder.ibm.com/infocenter/tivihelp/v2r1/index.jsp?topic=/com.ibm.itame.doc/welcome.htm

Questions & Feedback