

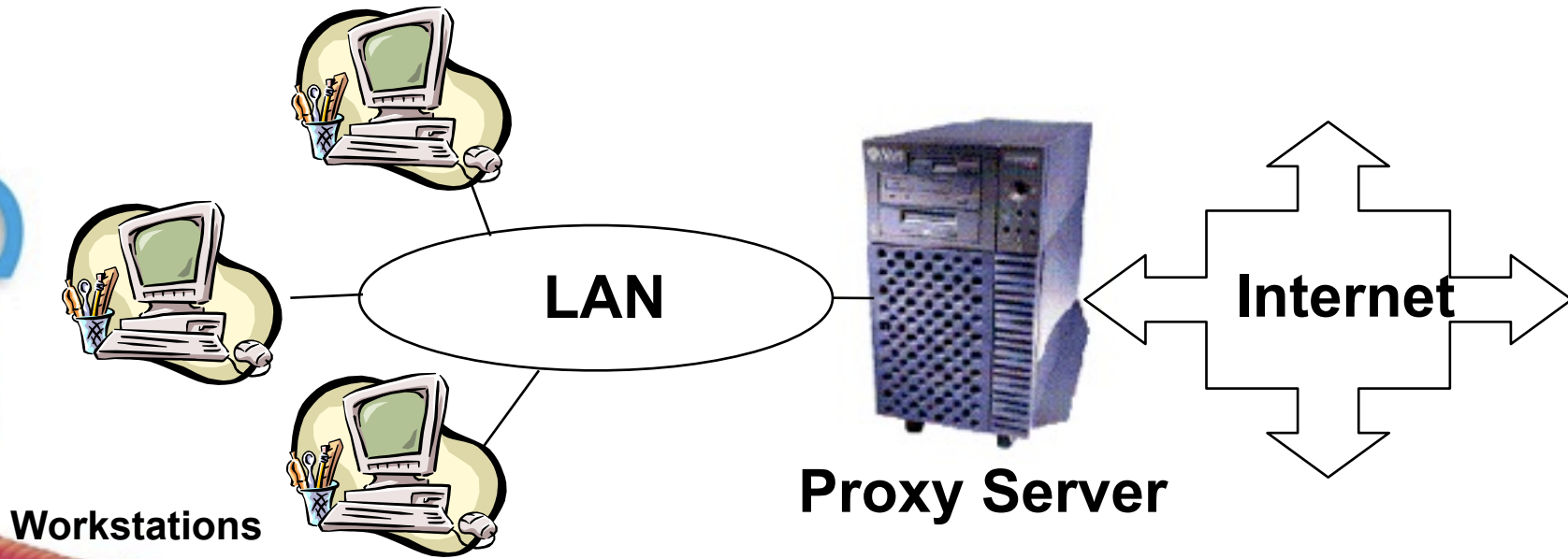
What we will cover

- Client-side proxy (Proxy Server)
- Server-side proxy (Reverse Proxy Server)
- Manipulating HTTP headers



Client-Side Proxy

- Desktop workstations not directly addressable from the outside world.
- Client-Side Proxy Server acts as the interface



Features of Proxy Server

- Performance
- Monitoring
- Filtering
- Caching (with mod_cache)



Enabling Proxy Support in Apache

- `LoadModule proxy_module mod_proxy.so`
- `LoadModule proxy_connect_module mod_proxy_connect.so`
- `LoadModule proxy_ftp_module mod_proxy_ftp.so`
- `LoadModule proxy_http_module mod_proxy_http.so`

- All above modules should be in there in `httpd.conf` by default



Proxy Server Directives:

- Describes what content is handled via the proxy server
- Allows for fine grained control over access, filters, etc...
- `<Proxy *>`
 - Order Deny,Allow
 - Deny from all
 - Allow from yournetwork.example.com`</Proxy>`



Proxy Server Directives:

- Describes which ports the CONNECT method is allowed to permit access to.
- By default, HTTPS (443) and SNEWS (563) are enabled
- Example:
 - AllowCONNECT 443 563 8443



Proxy Server Directive:

- Defines remote Proxy Servers for use by local Proxy Server
- Can specify a URL pattern
- Example:
 - `ProxyRemote * http://remote-server:3128`
- In conjunction with *ProxyDomain* and *NoProxy*, can be used to setup a network of departmental Proxy Servers cascaded through a corporate intranet, each forwarding external requests higher up the chain



Proxy Server Directives:

- Targets to which the Proxy will connect directly without using a *ProxyRemote*
 - Subnets (e.g. “192.168.1.0/21”)
 - IP addresses (e.g. “192.168.1.1”)
 - Hosts (e.g. “www.foo.com”)
 - Domains (e.g. “.foo.com”)
- Useful for Apache Proxy servers that reside within the intranet
- Example:
 - NoProxy .foo.dom 192.168.1.0/21



Proxy Server Directives:

- Sets a list of URL substrings, host names, and domain names, separated by spaces
- Proxy Server will block HTTP, HTTPS and FTP requests to above targets
- Example:
 - ProxyBlock cybersex *.sex.com



Proxy Server Directives:

- Sets the default domain for requests with incomplete hostnames.
- The specified domain is appended, and a redirect returned to the browser
- Example:
 - ProxyDomain .covalent.net



Proxy Server Directives:

- Enables client-side Proxying
- Default is off
- Example:
 - ProxyRequests on



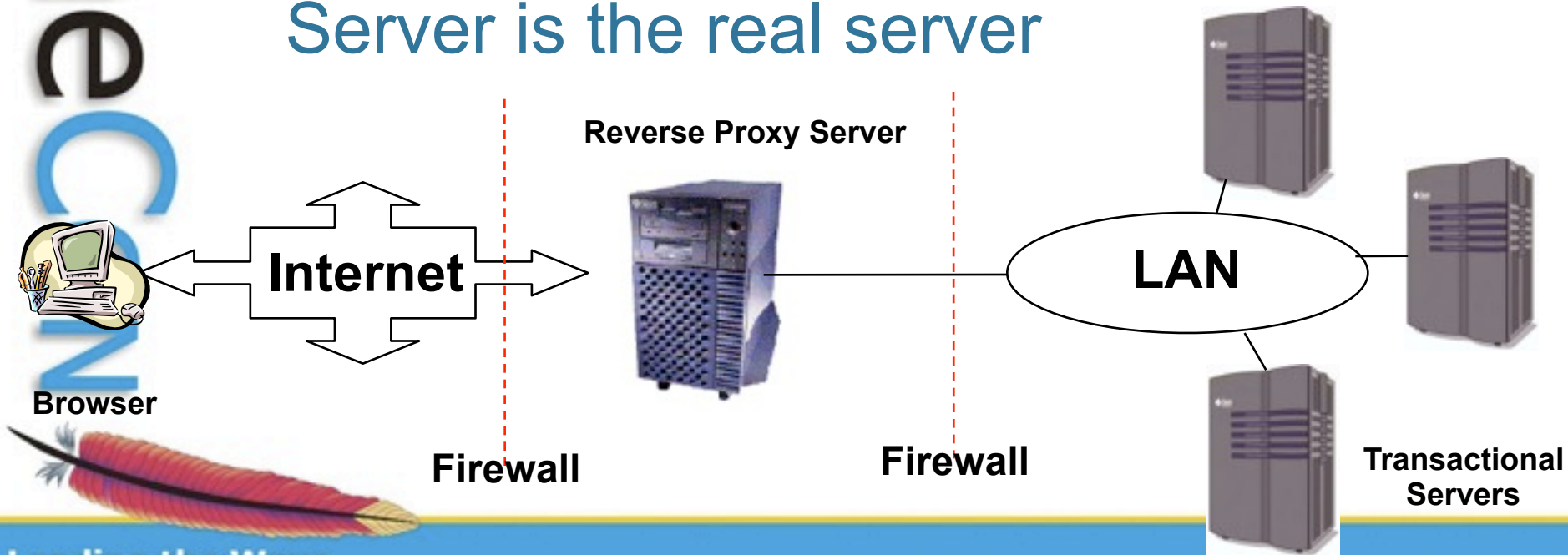
Proxy Server Directives:

- Controls how the server handles HTTP1.1 “Via:” headers.
- Default: off (“Via” header passes through unchanged)
- On: “Via” header field will be added
- Full: “Via” header field will have Apache server version info
- Block: All “Via” headers removed
- Don’t worry about this setting at all!
- Example:
 - ProxyVia Full



Server-Side Proxy (Reverse Proxy)

- Operated at the server end of the transaction
- Completely transparent to the Web Browser – thinks the Reverse Proxy Server is the real server



Features of Reverse Proxy

- Security
 - Uniform security policy can be administered
 - The real transactional servers are behind the firewall
- Delegation, Specialization, Load Balancing



Configuring Reverse Proxy

- Set ProxyRequests Off
- Apply ProxyPass, ProxyPassReverse and possibly RewriteRule directives



Reverse Proxy Directives:

- Allows remote server to be mapped into the space of the local (Reverse Proxy) server
- Example:
 - ProxyPass /secure/ <http://securerserver/cgi-bin/>
 - Presumably “securerserver” is inaccessible directly from the internet



Reverse Proxy Directives:

- Used to specify that redirects issued by the remote server are to be translated to use the proxy before being returned to the client.
- Syntax is identical to *ProxyPass*; used in conjunction with it
- Example:
 - `ProxyPass /secure/ http://secureserver/cgi-bin/`
 - `ProxyPassReverse /secure/ http://secureserver/cgi-bin/`

Manipulating HTTP Headers:

- Modify HTTP request and response headers
 - Can be used in Main server, Vhost, Directory, Location, Files sections
- Headers can be merged, replaced or removed
- Pass on client-specific data to the backend server
 - IP Address
 - Request scheme (HTTP, HTTPS)
 - UserAgent
 - SSL connection info
 - etc.
- Shield backend server's info from the clients
 - Strip out Server name
 - Server IP address
 - etc.



mod_headers directives:

- Header set|append|add|unset|echo *header* [value [env=[!]*variable*]]
 - set: replaces any prev header with this name
 - append: appended to any existing header of same name (name=val1,val2)
 - add: added to existing set of headers, even if same name exists (confusing)
 - unset: header removed
 - echo: Request headers echoed back in response headers
- Can use following specifiers in *value*:
 - %t: Time the request was received
 - %D: Duration of the request
 - %*{FOOBAR}*e: Contents of the env var FOOBAR



“Header” examples

- Copy all request headers that begin with “TS” to response headers
 - Header echo ^TS
- Say hello to Joe
 - Header add JoeHeader “Hello Joe!”
- Set header conditionally
 - If header “MyRequestHeader: value” is present, response will contain “MyHeader” header:

```
SetEnvIf MyRequestHeader value HAVE_MyRequestHeader
```

```
Header add MyHeader “%D %t mytext” env=HAVE_MyRequestHeader
```



mod_header directives:

- RequestHeader set|append|add|unset *header* [value]
 - set: replaces any prev header with this name
 - append: appended to any existing header of same name (name=val1,val2)
 - add: added to existing set of headers, even if same name exists (confusing)
 - unset: header removed
- Remember, sequence is important! Following will result in “MHeader” to be stripped from the response:
 - RequestHeader append MyHeader “value1”
 - RequestHeader append MyHeader “value2”
 - RequestHeader unset MyHeader



Example: Integration with App Servers

- Pass additional info about Client Browsers to the App Server:

```
ProxyPass / http://backend.covalent.net
```

```
ProxyPassReverse / http://backend.covalent.net
```

```
RequestHeader set X-Forwarded-IP %{REMOTE_ADDR}e
```

```
RequestHeader set X-Request-Scheme %{REQUEST_SCHEME}e
```



Example: Integration with App Servers

- App Server receives the following HTTP headers:
 - X-Forwarded-IP: 10.0.0.3
 - X-Request-Scheme: https

