Red Hat IT features:

Sidecar that authentication

Dustin Minnich - Principal Systems Administrator
Lauren Santiago - Senior Systems Administrator
Henry Graham - Senior Systems Administrator
Alec Henninger - Senior Software Applications Engineer
Lab Code: T93F1B
Activation Key: hunter2
Firefox ONLY
The Problem
Several people solving the same complex issue in different ways
IAM Services:

- LDAP
- Kerberos
- PKI
- OTP
- SSO

- PHP + SAML
- Java + PKI
- Ruby + OIDC
● Authentication protocols are complex
  ○ Takes time for IAM team members to master them
  ○ Devs don’t want to learn them
  ○ Vague with their MUSTs and MAYs

● Authentication protocol libraries differ in quality
  ○ Feature parity concerns
  ○ CVE concerns

● QA testing now relies on auth servers being up

● Diversity is a weakness
  ○ No DRY on an organizational level
  ○ Working session meetings required for integrations
  ○ Troubleshooting and support is harder
  ○ Several libraries must be audited
The Solution

A single well-supported abstraction layer that everybody uses for auth
• Time well spent
  ○ Write code that makes money instead of learning auth protocols
• Uniformity
  ○ Hands free automation is possible
  ○ Easier to support a known quantity
  ○ One framework to audit and keep patched
• Ability to test locally
The Lab Setup
The Code
<Location />
  Require valid-user
  AuthType "Mellon"
  MellonEnable "auth"
  MellonDecoder "none"
  MellonVariable "cookie"
  MellonSecureCookie On
  MellonUser "NAME_ID"
  MellonSessionDump Off
  MellonSamlResponseDump Off
  MellonEndpointPath "/secret/endpoint"
  MellonDefaultLoginPath "/"
  MellonSessionLength 86400
  MellonOrganizationURL "http://www.redhat.com"
  MellonSSPliveateKeyFile /etc/httpd/conf/mellon-signing.key
  MellonSPCertFile /etc/httpd/conf/mellon-signing.crt
  MellonIDPMetadataFile /tmp/metadata.xml
  MellonSamlResponseDump Off
  MellonSessionDump Off
  MellonMergeEnvVars On ";"
  RequestHeader unset username
  RequestHeader unset email
  RequestHeader unset first_name
  RequestHeader unset last_name
  RequestHeader unset groups
  RequestHeader unset header_security
  RequestHeader set username "%(MELLON_NAME_ID)" env=MELLON_NAME_ID
  RequestHeader set email "%(MELLON urn:oid:1.2.840.113549.1.9.1)" env=MELLON urn:oid:1.2.840.113549.1.9.1
  RequestHeader set first_name "%(MELLON urn:oid:2.5.4.42)" env=MELLON urn:oid:2.5.4.42
  RequestHeader set last_name "%(MELLON urn:oid:2.5.4.4)" env=MELLON urn:oid:2.5.4.4
  #RequestHeader set username "%(MELLON username)" env=MELLON username
  #RequestHeader set email "%(MELLON email)" env=MELLON email
  #RequestHeader set first_name "%(MELLON first_name)" env=MELLON first_name
  #RequestHeader set last_name "%(MELLON last_name)" env=MELLON last_name
  RequestHeader set groups "%(MELLON groups)" env=MELLON groups
  RequestHeader set header_security "test1234"
</Location>

<Location /sidecar>
  MellonEnable "off"
  Order Allow,Deny
  Allow from all
  Satisfy any
</Location>
public static class SideCarSecurityConfig extends WebSecurityConfigurerAdapter {
    private Environment environment;
    
    @Override
    protected void configure(AuthenticationManagerBuilder auth) throws Exception {
        PreAuthenticatedAuthenticationProvider preAuthProvider = new PreAuthenticatedAuthenticationProvider();
        // See AuthenticationDetails set below
        preAuthProvider.setPreAuthenticatedUserDetailsService(new HeaderBasedUserDetailsService());
        auth.authenticationProvider(preAuthProvider);
    }
    
    @Override
    protected void configure(HttpSecurity http) throws Exception {
        RequestHeaderAuthenticationFilter headerAuth = new RequestHeaderAuthenticationFilter();
        headerAuth.setAuthenticationDetailsSource((req) -> new ServerWebExchangeHttpSessionWithHttpServletRequest(req));
        headerAuth.setPrincipalDetailsSource((req, authDetails) -> new HeaderBasedAuthenticationDetails(req, environment.getProperty("principal.request-header", "username"), environment.getProperty("principal.request-header", "password"), Collections.emptyList()));
        headerAuth.setAuthenticationManager(new HeaderBasedAuthenticationManager());
    }
    
    private static class HeaderBasedUserDetailsService implements UserDetailsService {
        public UserDetails loadUserDetails(PreAuthenticatedAuthenticationToken token) throws UsernameNotFoundException {
            return new UsernamePasswordAuthenticationToken(token); // TODO replace with real implementation
        }
    }

    private static class HeaderBasedAuthenticationDetails implements UserDetails {
        public String getUsername() {
            return environment.getProperty("principal.request-header", "username");
        }
        public Collection<? extends GrantedAuthority> getAuthorities() {
            return environment.getProperty("principal.request-header", "authorities").split(",");
        }
    }
}

#redhat #rhsummit
Less Talking.
More computing.
Questions?
FINAL TASK:
FILL IN EVALUATION FORM IN RED HAT SUMMIT APP, DO IT NOW!

Red Hat IT features: Sidecar that authentication

Dustin Minnich - Principal Systems Administrator
Lauren Santiago - Senior Systems Administrator
Henry Graham - Senior Systems Administrator
Alec Henninger - Senior Software Applications Engineer
Visit us at the “Ask Me Anything” booth and ask us about how we implement and deploy Red Hat products!

redhat.com/redhat-on-redhat
<table>
<thead>
<tr>
<th>Date</th>
<th>Time</th>
<th>Room</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>Thurs, May 9</td>
<td>11:00 - 11:45am</td>
<td>Room 104AB</td>
<td>Red Hat on Red Hat: Transitioning Red Hat IT to hybrid cloud infrastructure using OpenStack and Ceph Storage</td>
</tr>
<tr>
<td>Thurs, May 9</td>
<td>3:15 - 4:00pm</td>
<td>Room 155</td>
<td>Developing and running cloud-native apps on OpenShift in Red Hat's IT organization</td>
</tr>
<tr>
<td>Thurs, May 9</td>
<td>3:15 - 4:00pm</td>
<td>Room 154</td>
<td>Developing and deploying applications in a multisite hybrid cloud</td>
</tr>
</tbody>
</table>