```
From 760bb358b8f53e52cf415888a4ac858fd99bb24e Mon Sep 17 00:00:00 2001
From: Robert Rothenberg <rrwo@cpan.org>
Date: Sun, 3 Aug 2025 13:37:07 +0100
Subject: [PATCH] Use Crypt::SysRandom to generate the session_id

The session_id generation was based on Plack::Middleware::Session, which
was insecure [CVE-2025-40923]. This was fixed in v0.35 by using a secure
source of random bytes.

This makes a similar changes.
---
 cpanfile                                |  4 ++--
 lib/Plack/Middleware/Session/Simple.pm | 27 +++++++++++++++------------
 2 files changed, 15 insertions(+), 16 deletions(-)

diff --git a/cpanfile b/cpanfile
index 871c733..39cb2e7 100644
--- a/cpanfile
+++ b/cpanfile
@@ -1,10 +1,11 @@
 requires 'perl', '5.008001';
 requires 'parent', '0.223';
-requires 'Digest::SHA1', '2.13';
+requires 'Crypt::SysRandom', 0;
 requires 'Cookie::Baker', '0.02';
 requires 'Plack', '1.0029';
 requires 'Scalar::Util';
 requires 'Tie::Hash';
+recommends 'Crypt::SysRandom::XS' => '0';

 on 'test' => sub {
     requires 'Test::More', '0.98';
@@ -14,4 +15,3 @@ on 'test' => sub {
     requires 'HTTP::CookieJar', '0.005';
     requires 'Test::Requires', '0.07';
 };
-
diff --git a/lib/Plack/Middleware/Session/Simple.pm b/lib/Plack/Middleware/Session/Simple.pm
index 706d038..826e6bb 100644
--- a/lib/Plack/Middleware/Session/Simple.pm
+++ b/lib/Plack/Middleware/Session/Simple.pm
@@ -4,7 +4,7 @@ use 5.008005;
 use strict;
 use warnings;
 use parent qw/Plack::Middleware/;
-use Digest::SHA1 qw//;
+use Crypt::SysRandom qw/random_bytes/;
 use Cookie::Baker;
 use Plack::Util;
 use Scalar::Util qw/blessed/;
@@ -40,7 +40,7 @@ sub prepare_app {

     if ( !$self->sid_generator ) {
         $self->sid_generator(sub{
-            Digest::SHA1::sha1_hex(rand() . $$ . {} . time)
+            unpack('H*', Crypt::SysRandom::random_bytes(20));
         });
     }
     if ( !$self->sid_validator ) {
@@ -58,7 +58,7 @@ sub call {

     my $tied;
     if ($id && $session) {
-        $tied = tie my %session,
```

```
+        $tied = tie my %session,
             'Plack::Middleware::Session::Simple::Session', %$session;
         $env->{'psgix.session'} = \%session;
         $env->{'psgix.session.options'} = {
@@ -66,7 +66,7 @@ sub call {
         };
     } else {
         my $id = $self->{sid_generator}->();
-        $tied = tie my %session,
+        $tied = tie my %session,
             'Plack::Middleware::Session::Simple::Session';
         $env->{'psgix.session'} = \%session;
         $env->{'psgix.session.options'} = {
@@ -122,7 +122,7 @@ sub finalize {
             $options->{id} = $self->{sid_generator}->();
             my $val = $session->[0];
             $val = $self->{serializer}->[0]->($val) if $self->{serializer};
-            $self->{store}->set($options->{id}, $val);
+            $self->{store}->set($options->{id}, $val);
         } else {
             my $val = $session->[0];
             $val = $self->{serializer}->[0]->($val) if $self->{serializer};
@@ -133,10 +133,10 @@ sub finalize {
     if ( $set_cookie ) {
         if ($options->{expire}) {
             $self->_set_cookie(
-                $options->{id}, $res, %$options, expires => 'now');
+                $options->{id}, $res, %$options, expires => 'now');
         } else {
             $self->_set_cookie(
-                $options->{id}, $res, %$options);
+                $options->{id}, $res, %$options);
         }
     }
 }
@@ -155,7 +155,7 @@ sub _set_cookie {
         $options{expires} = $self->{expires};
     }

-    my $cookie = bake_cookie(
+    my $cookie = bake_cookie(
         $self->{cookie_name}, {
             value => $id,
             %options,
@@ -217,7 +217,7 @@ Plack::Middleware::Session::Simple - Make Session Simple
         my $counter = $env->{'psgix.session'}->{counter}++;
         [200,[], ["counter => $counter"]];
     };
-
+
     builder {
         enable 'Session::Simple',
             store => Cache::Memcached::Fast::Safe->new({servers=>[..]}),
@@ -230,7 +230,7 @@ Plack::Middleware::Session::Simple - Make Session Simple

 Plack::Middleware::Session::Simple is a yet another session management module.
 This middleware has compatibility with Plack::Middleware::Session by
-supporting psgix.session and psgi.session.options.
+supporting psgix.session and psgi.session.options.
 You can reduce unnecessary accessing to store and Set-Cookie header.

 This module uses Cookie to keep session state. does not support URI based session state.
@@ -266,8 +266,8 @@ If disabled, Plack::Middleware::Session::Simple does not output Set-
Cookie heade
             [200,[], ["login"]];
```

```
          },
      };
 -
 -    my $res = $app->(req_to_psgi(GET "/")); #res does not have Set-Cookie
 +
 +    my $res = $app->(req_to_psgi(GET "/")); #res does not have Set-Cookie
      my $res = $app->(req_to_psgi(GET "/login")); #res has Set-Cookie

  If you have a plan to use session_id as csrf token, you must not disable keep_empty.
 @@ -315,7 +315,7 @@ serialize,deserialize method. Optional. This is useful with
 Cache::FastMmap
      $app;
    };

 -=back
 +=back

  =head1 LICENSE

 @@ -329,4 +329,3 @@ it under the same terms as Perl itself.
  Masahiro Nagano E<lt>kazeburo@gmail.comE<gt>

  =cut
 -
```