AAI@EduHr - spremanje lozinki u obliku SSHA (salted SHA)

Lozinke u OpenLDAP-u službenom paketu za sustav AAI@EduHr su enkodirane kao SHA, koji se smatra zastario i nesiguran, a uz to je podložan tzv. "*dictionary attack*" napadu. Na internetu postoje i razni *online* dekoderi za SHA nizove. Zato bi bilo bolje lozinke spremati enkodirane u SSHA formatu sa dodanim "začinom" (***salt***). Salt je slučajan niz znakova koje se ukodirava u već kodiranu korisničku lozinku, te čak i u slučaju korištenja iste lozinke sam SSHA zapis će drugačiji. Time postaje otporan na "*dictionary attack*", jer ista enkodirana lozinka više ne izgleda isto kao i na nekom drugom sustavu .

Kako bi to postigli, potrebno je urediti datoteku */usr/lib/aosi/AOSICFG.pm.* **Uređivanjem te datoteke trebate imati na umu da dok se isto ili slično ne implemetira u AOSI, svakom ćete nadogradnjom ovu funkcionalnost izgubiti (barem dok ponove ne uredite datoteku).**

Promjena je oko linije 2310, pa nadalje. Isprobana je zamjena lozinke preko AOSI web sučelja i sve radi u redu. Login na sve servise sa SSHA lozinkom u LDAP-u također radi bez greške.

Smatram da bi se rečeno trebalo ubaciti u službeni AOSI, jer je SSHA enkodiranje lozinke sa slučajnim "salt" nizom puno sigurnija opcija čuvanja lozinki od "čistog" SHA (koji je podložan *dictionary* napadu).

Naime, podaci u LDAPu sa AAI podacima se koriste i kao podaci za mail server (korisničko ime i lozinka), koji se sinhroniziraju s drugim LDAP-om, koji onda služi mailserveru. Primjera radi, do sada smo imali 3 "probijene" lozinke, pa bi da nemamo različite lozinke za AAI i za mail bili puno izloženiji daljnjim napadima. Iz tog razloga ovaj članak služi kao prijedlog za ovu promjenu, a sve u svrhu povećanja sigurnosti sustava AAI@EduHr.

```
########################################################################
# Encodes input string in SHA algorithm if it is not encoded allready
# Input: string to encode
# Output: encoded string
########################################################################
sub encode_pwd {
    my ($in) = @_;
    my ($out, $log);
    # 22.01.2015 dodan jedan red
    my $salt = shift || &make_salt();
    $log = (caller(0))[3];

   #AOSICFG::write_ws_log("| $log(in): $in;") if $config::aosi_debug_level > 3;
   if ($in =~ /^\{SHA\}.*/) {
       $out = $in;
   }
   else {
       #originalna linija $out = '{SHA}' . sha1_base64($in) . '=';
       #22.01.2015 zamijenjena sa linijom niže
       $out = &ssha( $in, $salt )
   }
   AOSICFG::write_ws_log("| $log(out): $out;") if $config::aosi_debug_level > 3;
   return $out;
```

```
}

# 22.01.2015. dodane funkcije ssha i make_salt
sub ssha {
    my $pw = shift;
    my $salt = shift || &make_salt();
    return "{SSHA}" . MIME::Base64::encode( Digest::SHA::sha1( $pw . $salt ) . $salt,
 '' );
}

sub make_salt {
    my $len = 8 + int( rand(8) );
    my @bytes = ();
    for my $i ( 1 .. $len ) {
        push( @bytes, rand(255) );
    }
return pack( 'C*', @bytes );
}
```

pon, 2015-01-26 14:29 - Krešimir Mihalj**Vijesti:** Sigurnosni propusti [1]
Kuharice: Linux [2]
Vote: 0

No votes yet

Source URL: https://sysportal.carnet.hr/node/1499

Links
[1] https://sysportal.carnet.hr/taxonomy/term/14
[2] https://sysportal.carnet.hr/taxonomy/term/17