Telefónica Argentina

Diseño de OTP en NetScaler

Diseño de OTP en Citrix® NetScaler™

Junio 2014

Preparado Por:

Daniel Castro
Citrix Consulting

Historia de versiones

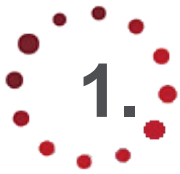
Versión	Descripción del Cambio	Hecho por	Fecha
0.1	Creación del documento	Fabián Baena.	15/08/2012
0.1.1	Creación del template AGEE	Daniel Castro	01/08/2013
0.2	Modificación OTP	Daniel Castro	13/06/2014
0.3	QA	Diego Madiedo	19/06/2014

Firmas del entregable

Las firmas a continuación indican el acuerdo entre Telefónica Argentina y Citrix Consulting sobre el contenido del Diseño de OTP en Citrix® NetScaler™.

Telefonica Argentina	Citrix Consulting
Nombre: Martin Giller	Nombre: Daniel Castro
Firma:	Firma:
Titulo:	Titulo: Principal Consultant
Fecha:	Fecha: Junio 13 de 2014

Índice



1. Vista General

1.1. Resumen Ejecutivo

Uno de los requerimientos de seguridad de Telefónica Argentina definen que todos los usuarios y proveedores que acceden remotamente a la plataforma realicen un proceso de autenticación de dos factores.

Actualmente Telefónica Argentina tiene una solución hecha a la medida como segundo factor de autenticación, esta solución tiene el nombre de MoviID. Esta solución tiene una alta complejidad ya que fue desarrollada hace muchos años por personal que ya no hace parte de Telefónica. Adicionalmente tiene componentes de software de una alta complejidad inerte sin tener en cuenta que la interacción con el usuario no es fácil.

Para que un usuario haga uso de MoviID, debe marcar desde su teléfono corporativo a un número especial asterisco (*). Allí debe marcar su clave telefónica y el sistema le va a dictar por medio de una grabación su token temporal. Por el contrario el nuevo sistema (aquí descrito) usa como clave los últimos cuatro dígitos del numero telefónico del usuario y el token es enviado al usuario por medio de un SMS.

Telefónica Argentina se encuentra en el proceso de actualización de su plataforma, para esto ha instalado unos equipos NetScaler para el acceso remoto. Adicionalmente se está remplazando la solución de segundo factor de autenticación. Este documento busca explicar un desarrollo a la medida hecho por Citrix Systems para Telefónica Argentina.

Este desarrollo busca sustituir MoviID por una nueva solución que debe ser más sencilla para los usuarios y debe estar soportada por dispositivos móviles que ingresen usando Citrix Receiver.

1.2. Vista General del Entregable

Este documento de diseño de One Time Password (OTP) para NetScaler es el resultado de la colaboración con el área de Seguridad Estratégica de Telefónica Argentina y Citrix Consulting.

Basado en las discusiones colaborativas, la arquitectura descrita en este documento representa las decisiones de diseño tomadas en conjunto con Citrix Consulting durante el transcurso de este encuentro.

Este diseño arquitectónico está organizado de la siguiente manera:

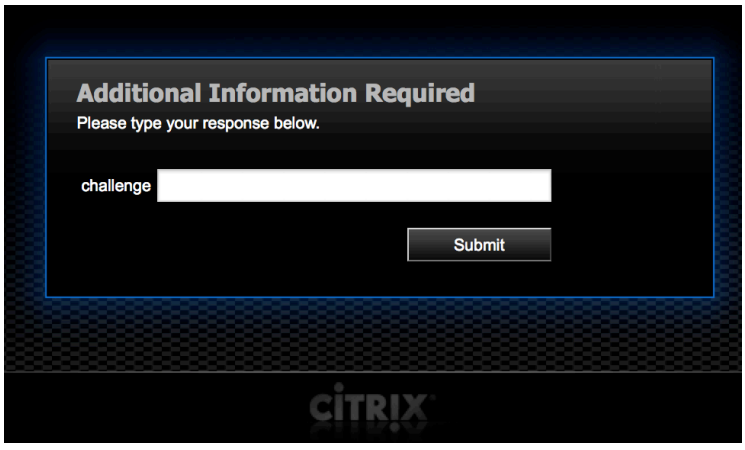
Sección	Temas cubiertos

Sección	Temas cubiertos
Vista General	Esta sección provee una vista general del proyecto, incluyendo temas de diseño. Se detallan los fundamentos de arquitectura del diseño de NetScaler.
Diseño de la Solución	Detalle de cada uno de los compontes que se involucran en la solución.
Código de la Solución	Muestra el código fuente del componente Servidor
Mantenimiento	Dependencias e Instrucciones de instalación y los diferentes archivos involucrados.
Soporte	Detalla el proceso de soporte, el proceso de escalación con Citrix Consulting.

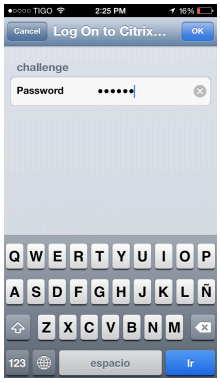
Cambios en la plataforma actual

Desde la perspectiva de los usuarios de Telefónica no existe ningún cambio en la configuración de NetScaler ya que en la actualidad se hace uso de un segundo factor de autenticación.

Sin embargo para los usuarios si va a existir una pantalla adicional denominada "Challenge" que hace parte de las funcionalidades de NetScaler. Esta pantalla es desplegada después de la pantalla de login tradicional cuando el usuario ingreso los datos correctamente. Para los usuarios ingresando vía Web se ve de la siguiente manera:



Esta página tiene localización de idioma por defecto por lo cual se va a desplegar correctamente en español. Si el usuario esta ingresando desde un dispositivo móvil, por ejemplo iPhone se ve de la siguiente manera en el Citrix Receiver:



Toda la configuración de NetScaler Gateway incluyendo autenticación por dominio activo y MoviID fue realizada en un proyecto anterior realizado por DWS.

La única configuración adicional corresponde al Servidor RADIUS desarrollado por parte de Citrix (ver capitulo 1.7), y su correspondiente política de autenticación secundaria en el virtual server de NetScaler Gateway, esta configuración se puede ver a mas detalle en 1.6.

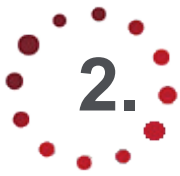
1.4. Beneficios y necesidades que resuelve la solución

Para Telefónica Argentina la solución actual de MoviID tiene dos problemáticas para los usuarios que no tienen solución:

1. Los usuarios no pueden marcar a MoviID por fuera de la argentina.
2. Al ingresar desde un dispositivo móvil, se debe usar el mismo teléfono para ingresar vía Citrix Receiver y hacer la llamada a MoviID.

Ya que la nueva plataforma de OTP utiliza SMS como mecanismo para entregar el token del usuario este puede estar en cualquier lugar del mundo, tan solo tiene que estar usando su roaming internacional. Además el token enviado en el mensaje SMS permite copiar y ser pegado directamente en la pantalla de Citrix Receiver que se mostro anteriormente aumentando en gran medida la facilidad de su uso para el usuario

Desde la perspectiva de negocio para Telefónica Argentina MoviID es una plataforma legacy sin soporte alguno, la nueva plataforma OTP de Citrix cuenta con el soporte por parte de DWS.



2. Diseño de la solución

1.5. Requerimientos funcionales

La solución a la medida de OTP para NetScaler tiene un único componente de software que hace toda la loica necesaria para realizar el proceso autenticación. Los siguiente son los requerimientos funcionales que debe cumplir la solución:

1. Buscar el número telefónico de un usuario en los dos controladores de dominio; tasa y móviles. Primero se verifica contra tmoviles, luego contra tasa. Si no existe numero telefónico en ambos dominios se niega el acceso.

2. Para poder diferenciar dos usuarios idénticos en los dos dominios se debe verificar que la clave inicial del usuario corresponda a los últimos 4 dígitos del teléfono del usuario extraídos en el punto 1.

3. Si los últimos 4 dígitos corresponde se debe generar un código alfabético aleatorio de 6 caracteres que no correspondan a una palabra.

4. Se debe enviar una petición http al Gateway SMS que ya existen en Telefónica Argentina, de acuerdo a las especificaciones que provea Telefónica, donde además de un breve mensaje explicativo se debe incluir el código generado en el punto 3. Este código se denomina el token.

5. El usuario puede errar hasta 3 veces al ingresar con su token, al realizar el cuarto intento se debe invalidar la transacción y debe comenzar de nuevo

6. El token tiene un tiempo de vida máximo de 300 segundos, pasado este tiempo si el usuario intenta ingresar con un token expirado, debe volver a comenzar de nuevo.

7. El sistema soporta únicamente paquetes de petición tipo RADIUS "Access-Request". Para mayor información sobre el protocolo consultar el RFC2866 en http://tools.ietf.org/html/rfc2866

8. El sistema responde al NetScaler con paquetes tipo RADIUS Access-Accept, Access-Reject y Access-Challenge.

9. Existe una bitácora (log) del sistema donde se guarda cada acceso inicial del usuario, sus fallas de autenticación y sus accesos exitosos al sistema.

1.6. Configuración de NetScaler

A continuación se muestran las diferentes configuraciones necesarias para el correcto funcionamiento de la solución.

Politicas

Para poder implementar este esquema es necesario usar autenticación secundaria por cascada (más información en http://support.citrix.com/proddocs/topic/netscaler-gateway-101/ng-multifactor-auth-tsk.html), esto quiere decir que ambos sistemas (MoviID y OTP) están configurados simultáneamente. El usuario puede ingresar si alguno de los dos sistemas responde afirmativamente cuando se presenta el token del usuario.

Nombre	Expresión	Perfil	Prioridad	Bind
OTP_NS	ns_true	OTP_RADIUS	110	Virtual Server - Secondary

La política corresponde a una segunda política RADIUS en cascada.

Perfil

El perfil es el objeto en el NetScaler que contiene toda la información necesaria para poder utilizar el recurso, a continuación están las configuraciones necesarias:

Configuración	Valor	Justificación
Direccion IP	TBD	No es claro si el servidor radius va a ser instalado en el NetScaler o en algún servidor Linux proveido por Telefónica. Si es en el NetScaler se deben cumplir los requerimientos de librerías descritos en 1.7
Puerto	1812	Este puerto corresponde al puerto por defecto para servicios RADIUS
Shared Secret	mysecret	Este debe ser cambiado por Telefónica por un valor más seguro
Password Encoding	PAP	Se ha configurado el servidor para soportar únicamente este tipo de cifrado en el password.

1.7. Sevidor RADIUS OTP

Esta solución se basa en que el NetScaler solo puede comunicarse con el Servidor OTP por medio del protocolo RADIUS, basados en esto se ha creado un software que corre

en el Kernel FreeBSD del NetScaler o en un servidor Linux como un proceso iniciado por el sistema.

Para que dicho software pueda correr en el NetScaler es necesario cumplir con una serie de dependencias de software, la siguiente es la lista de librerías que deben ser instaladas en el sistema:

- mach/
    - auto/Net /
        - Gen/
            - EOF.al
            - FETCH.al
            - GETC.al
            - Gen.bs
            - Gen.so
            - PRINTF.al
            - READ.al
            - READLINE.al
            - RECV.al
            - STORE.al
            - TIEHANDLE.al
            - TIESCALAR.al
            - WRITE.al
            - _findxopt.al
            - _getxopt.al
            - _setxopt.al
            - accept.al
            - autosplit.ix
            - bind.al
            - delparam.al
            - didlisten.al
            - fcntl.al
            - fhvec.al
            - fileno.al
            - format_addr.al
            - format_local_addr.al
            - format_remote_addr.al
            - getfh.al
            - getlines.al
            - getropt.al
            - getsopt.al
            - ioctl.al
            - listen.al
            - new_from_fh.al
            - select.al
            - sendto.al
            - setdebug.al
            - setparam.al
            - setropt.al

```
                    •   setsopt.al
                    •   unbind.al
                ▪   Inet
                    •   Autosplit.ix
                    •   Bind.al
                    •   Setdebug.al
                    •   Unbind.al
                ▪   UDP
                    •   _setbuf_unbuf.al
                    •   autosplit.ix
                    •   PRINT.al
                    •   READLINE.al
                ▪   UNIX
                    •   autosplit.ix
                    •   bind.al
                    •   setdebug.al
        o   Net/
                ▪   Gen.pm
                ▪   Inet.pm
                ▪   TCP/
                    •   Server.pm
                ▪   TCP.pm
                ▪   UDP.pm
                ▪   UNIX/
                    •   Server.pm
                ▪   UNIX.pm
•   MD5.pm
•   RADIUS/
        o   Dictionary.pm
        o   Packet.pm
```

Al copiar las anteriores dependencias a la carpeta: /usr/local/lib/perl5/site_perl/5.8.8/ se puede ejecutar el servidor. Lo anterior se encuentra almacenado en archivo Tar comprimido en la carpeta /var, y se descomprime y ejecuta de manera automática por un script de inicio en el NetScaler.rc.

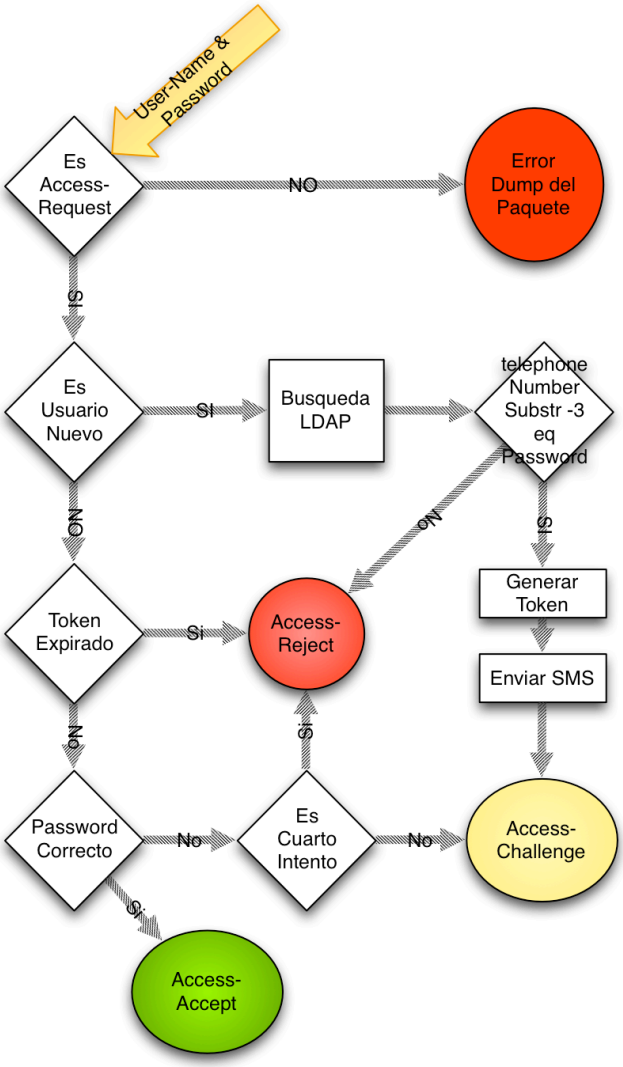
Las líneas correspondientes en el archivo RC son:

- tar –zxf radius.tar.gz -C /
- perl /var/captcha/radius.pl &

Si por el contrario se decide correr el servidor en una maquina Linux se debe instalar Perl 5.8.8 o superior en el sistema, las dependencias son instaladas por defecto por CPAN (el manejador de módulos de perl).

Algoritmo del Servidor

A continuación se presenta el diagrama de flujo del servidor:



A continuación se explican las tres respuestas que puede entregar el sistema al recibir un Access-Request.

1. Access-Reject envía al usuario de nuevo a la pagina inicial de login, solicitando de nuevo sus credenciales del Dominio

2. Access-Challenge envía al usuario a la pagina donde debe ingresar su token, esta pagina solo se puede ingresar posterior a las credenciales del dominio y haber indicado los últimos cuatro dígitos del teléfono

3. Access-Accept permite el ingreso a la plataforma

Codigo Servidor

A continuación se presenta el código del servidor en su estado a la fecha:

```
#!/usr/local/bin/perl -w

    use RADIUS::Dictionary;
    use RADIUS::Packet;
    use Net::Inet;
    use Net::UDP;
    use Net::LDAP;
    use Fcntl;
    use strict;

    # This is a VERY simple RADIUS authentication server which responds
    # to Access-Request packets with Access-Accept.  This allows anyone
    # to log in.
    my %usuarios;
    #$my @user = ("ABCDEF" ,time()+300 );
    my $username = "nata";
    #$usuarios{$username} = [@user];
    $usuarios{$username}[0] = "ABCDEF";
    $usuarios{$username}[1] = time()+300;
    $usuarios{$username}[2] = 0;
    my $string = "";
    my $time = 0;
    my $secret = "mysecret";  # Shared secret on the term server

    #lo necesario para LDAP
    my $ldap = Net::LDAP->new( '192.168.254.133' ) or die "$@";
    my $mesg = $ldap->bind( 'cn=admin',
                    password => 'ttikyy09'
                  );

   my $base = "dc=test, dc=com";
   my $attrs = [ 'cn','mail','TelephoneNumber' ];

    # Parse the RADIUS dictionary file (must have dictionary in current dir)
    my $dict = new RADIUS::Dictionary "dictionary"
      or die "Couldn't read dictionary: $!";

    # Set up the network socket (must have radius in /etc/services)
    my $s = new Net::UDP { thisservice => "radius" } or die $!;
    $s->bind or die "Couldn't bind: $!";
    $s->fcntl(F_SETFL, $s->fcntl(F_GETFL,0) | O_NONBLOCK)
      or die "Couldn't make socket non-blocking: $!";

    # Loop forever, recieving packets and replying to them
    while (1) {
      my ($rec, $whence);
      # Wait for a packet
      my $nfound = $s->select(1, 0, 1, undef);
      if ($nfound > 0) {
        # Get the data
        $rec = $s->recv(undef, undef, $whence);
        # Unpack it
        my $p = new RADIUS::Packet $dict, $rec;
        if ($p->code eq 'Access-Request') {
          # Print some details about the incoming request (try ->dump here)
          print $p->attr('User-Name'), " logging in with password ",
                $p->password($secret), "\n";
          #$username = $p->attr('User-Name');
          # Create a response packet
          my $rp = new RADIUS::Packet $dict;
```

```
        #print "Current User ".$username." ".$usuarios{$username}[0]." ".$usuarios{$username}[1]."\n";
        if ($usuarios{$p->attr('User-Name')})
        {
          #usuario ya registrado, es respuesta al challenge
          if($usuarios{$p->attr('User-Name')}[1]>time())
          {
            #verificación para ver si el token no esta expirado
            if($p->password($secret) eq $usuarios{$p->attr('User-Name')}[0])
            {
              #si el password corresponde al token
              print "USER OK".$p->attr('User-Name')."\n";
              $rp->set_code('Access-Accept');
              delete $usuarios{$p->attr('User-Name')};
            }else
            {
              #token y password no corresponden
              print "REJECT PASSWD MAL ".$p->attr('User-Name')." Numero de intentos fallidos: ".$usuarios{$username}[2]."\n";
              #aumenta el contador de errores.
              if($usuarios{$username}[2]>3)
              {
                $rp->set_code('Access-Reject');
                delete $usuarios{$p->attr('User-Name')};
              }else{
                $usuarios{$username}[2]++;
                $rp->set_code('Access-Challenge');
              }

            }
          }else
            {
            #token expirado
            print "REJECT TOKEN EXPIRADO ".$p->attr('User-Name')."; Tiempo ahora:".time()."; Esperaba: ".$usuarios{$p-
>attr('User-Name')}[1]."\n";
            $rp->set_code('Access-Reject');
            delete $usuarios{$p->attr('User-Name')};
            }
          }
        else{
          #Si el usuario no esta registrado
          #buscar en LDAP telefono para verificar el los ultimos 4 digitos del telefono
          my $filter = "uid=".$p->attr('User-Name');
            $mesg = $ldap->search ( base    => "$base",
                        scope   => "sub",
                        filter  => $filter,
                        attrs   =>  $attrs
                      );

            print "MSG: ".$mesg->code."\n";

            my $entry;
            foreach $entry ($mesg->entries) {
              print "DN=".$entry->dn()."\n";
              if(!$entry->exists("TelephoneNumber"))
              {
                print "No hay Telefono registrado en LDAP\n";
                $rp->set_code('Access-Reject');
              }else{
                print "Phone: ".$entry->get_value("telephoneNumber")."\n";
                my $phone = $entry->get_value("telephoneNumber");
                my $phone = substr($phone, -3);
                if($p->password($secret) eq $phone)
                {
                  #crear token
                  for (0..5) { $string .= chr( int(rand(25) + 65) ); } print $string."\n";
                   $usuarios{$username}[0] = $string;
```

```
                #crear tiempo de expiracion
                $time = time()+300;
                $usuarios{$username}[1] = $time;
                $usuarios{$username}[2] = 0;
                #print "New User with token ".$string." expires at ".$time."\n";
                $string = "";
                $time = 0;
                #print "Current User ".$username." ".$usuarios{$username}[0]." ".$usuarios{$username}[1]."\n";
                $rp->set_code('Access-Challenge');
            }
            }
        }
      }
     $rp->set_identifier($p->identifier);
     $rp->set_authenticator($p->authenticator);
     # (No attributes are needed.. but you could set IP addr, etc. here)
     # Authenticate with the secret and send to the server.
     $s->sendto(auth_resp($rp->pack, $secret), $whence);
    }
    else {
     # It's not an Access-Request
     print "Unexpected packet type recieved.";
     $p->dump;
    }
   }
}
```

Para ver la ultima versión puede obtener el código fuente de:

https://github.com/evildani/OTP_NetScaler/blob/master/simple.pl

3. Operaciones

1.8. Mantenimiento

El archivo otp.tar.gz contiene todos los archivos necesarios para correr el servidor, en caso de falla el principal síntoma será la negación de acceso a todos los usuarios. Esto se debe a que el sistema va a negar las peticiones de Token. Este debe auto instalarse tras cada reinicio de las máquinas sin necesidad de interactuar con el administrador. Es necesario verificar que el archivo /nsconfig/netscaler.rc contiene las líneas descritas en 1.7

Si el sistema no está entregando los mensajes de texto con el token, puede deberse a un problema en el Gateway de SMS, o a que la configuración para envío de SMS ha cambiado. Si lo segundo ha sucedido es necesario examinar el código del servidor y cambiar los parámetros del POST en la sección de SMS.

El servidor deja una bitácora en /var/log/token_radius.log donde se guardan el número telefónico al cual se le envió el token, el usuario y la respuesta del Gateway SMS. En otra entrada verá el acceso al sistema por parte del usuario, es decir cada ingreso debe tener dos entradas.

Para deshabilitar el sistema, se deben quitar las políticas de autenticación secundarias y se debe quitar del nivel global la política de responder POL_START_GOTO_STEP1. Con esas dos operaciones el NetScaler permite el ingreso directo de los usuarios al Virtual Server de NetScaler Gateway y la autenticación puede proceder contra Active Directory.

1.9. Soporte

El soporte de primer nivel siempre estará a cargo de DWS, por medio de la documentación y transferencia de conocimiento en sitio al personal de soporte de DWS que estará en la capacidad de atender pequeños incidentes sobre la plataforma.

En caso de que Telefónica requiera de hacer una actualización de su plataforma más allá de la versión 10.1 se debe coordinar con Citrix Systems una visita programada para poder estudiar el impacto que dicha actualización pueda tener sobre el sistema.

4. Apéndices

1.10. Apéndice A –

Pendiente, copiar líneas relevantes de configuración

851 West Cypress Creek Road Fort Lauderdale, FL 33309 954-267-3000 http://www.citrix.com