



NITROKEY - PIV

The specifications and information in this document are subject to change without notice. Companies, names, and data used in examples herein are fictitious unless otherwise noted. This document may not be copied or distributed by any means, in whole or in part, for any reason, without the express written permission of RCDevs.

Copyright (c) 2010-2017 RCDevs SA. All rights reserved.

<http://www.rcdevs.com>

WebADM and OpenOTP are trademarks of RCDevs. All further trademarks are the property of their respective owners.

Limited Warranty

No guarantee is given for the correctness of the information contained in this document. Please send any comments or corrections to info@rcdevs.com.

Nitrokey - PIV

[nitrokey PIV](#)

Authentication with a Nitrokey / PIV

In this How-To we will configure a user in WebADM for using a PIV key. We need a WebADM server already configured.

1. Import the Inventory

We need to create an inventory file like this:

```
"Type","Reference","Description","DN","Data","Status"  
"PIV Device","<ID1>","PIV Nitrokey","","PublicKey=<pub_key1>","Valid"  
"PIV Device","<ID2>","PIV Nitrokey","","PublicKey=<pub_key2>","Valid"  
"PIV Device","<ID3>","PIV Nitrokey","","PublicKey=<pub_key3>","Valid"
```

For my test, I have a Nitrokey Start with a PIV certificate and I use `gpg2 --card-edit` for the management of the Nitrokey. Please follow this documentation [Nitrokey - Installation](#).

We need to extract the public key. I do it with `pkcs15-tool`:

```
-bash-4.2# pkcs15-tool --read-public-key 03  
Using reader with a card: Nitrokey Nitrokey Start  
-----BEGIN PUBLIC KEY-----  
MIIBIjANBgkqhkiG9w0BAQEFAA0CAQ8AMIIBCgKCAQEAWiBZ8g4yHliKPSr/Kg4E  
cAJLHch+Kh6w6emzn9ZRxSfrBofS045x17oi7UsG80IrBRMIVTgX0zqMbTwnnPjk  
pep9dKe4FHEMaPEvNYhAwHDMGVhbYBcf7Ru3CsCM9NPqmbjeV/+zGsMxq8XbZLKP  
doW4EjtneTpqD8ummip1ZBTuaFXGi3D/SDxAWTy3DlA+QtU5E2HpU7tZghi5ygiy  
9przQct/pMCNX8WJgkLC58g/UtnVeClkh2GGalFrODR2hY0lhWQYhzNH5FzIBmEE  
NcPucSwB7/r0abV9hdW52qWXECGBIjKAXrA16n/4QsFJNlPJaysl5Pv4ZBqM86jo  
gwIDAQAB  
-----END PUBLIC KEY-----
```

We can create a file called `nitrokey.csv` with the serial number as ID and the right public key:

```
"Type","Reference","Description","DN","Data","Status"  
"PIV Device","67090940","PIV  
NitroKey","","PublicKey=MIIBIjANBgkqhkiG9w0BAQEFAA0CAQ8AMIIBCgKCAQEAWiBZ8g4yHliKPSr/Kg4Ec
```

We import the file. Under the the `Import` tab, we click on `Import Inventory File`:

□

We choose the `nitrokey.csv` file and click on `Import` :

□

□

Now, the PIV key is present in the inventory:

□

2. Assign the Nitrokey

We select the user in the LDAP tree on the left and add a `UNIX Account` extension:

□

We click on `Proceed` :

□

We `Extend Object` :

□

We click on `SSH Public Key Server` :

□

We click on `Register / Unregister SSH Key` :

□

We select `Register a hardware key (Inventoried)` , enter the `Serial Number` (Reference) and `Register` :

□

□

Now, the PIV key is well registered.

□

3. Test with SSH

We'll try with a CentOS 7 as an ssh server.

We install and configure `spankey_client` on it:

```
[root@centos7-client ~]# yum install
https://www.rcdevs.com/repos/redhat/rcdevs_release-1.0.0-0.noarch.rpm
[root@centos7-client ~]# yum clean all
[root@centos7-client ~]# yum install spankey_client -y
[root@centos7-client ~]# /opt/spankey/bin/setup
Enter one of your running WebADM node IP or hostname []: 192.168.3.236
Do you want to enable SpanKey Client for OpenSSH server (y/n)? [N]: y
Do you want to enable SpanKey Client NSS plugin (y/n)? [Y]:
Do you want to register SpanKey Client logrotate script (y/n)? [Y]:
Do you want SpanKey Client to be automatically started at boot (y/n)? [Y]:

    Primary OpenOTP service URL is: 'https://192.168.3.236:8443/spankey/'
    Secondary OpenOTP service URL is: 'NONE'
    Enable SpanKey Client for OpenSSH server: 'YES'
    Enable SpanKey Client NSS plugin: 'YES'
    Register SpanKey Client logrotate script: 'YES'
    SpanKey Client must be automatically started at boot: 'YES'

Do you confirm (y/n)?: y

Applying SpanKey Client settings from default configuration files... Ok
Retrieving WebADM CA certificate from host '192.168.3.236'... Ok
The setup needs now to request a signed 'SpanKey' client certificate.
This request should show up as pending in your WebADM interface and an administrator
must accept it.
Waiting for approbation... Ok
Updating entry 'client_id' in file '/opt/spankey/conf/spankey.conf'... Ok
Updating file '/etc/ssh/sshd_config'... Ok
Updating file '/etc/nsswitch.conf'... Ok
Updating file '/etc/pam.d/password-auth'... Ok
Registering SpanKey Client service...
Registering SpanKey Client service... Ok
Adding logrotate script... Ok

SpanKey Client has successfully been setup.

IMPORTANT: Do not forget to perform the following actions before you exit this session:
- Start SpanKey (/opt/spankey/bin/spankey start)
- Restart 'sshd'
- Restart 'nscd'

[root@centos7-client ~]#
```

For the ssh client, we use a mac mini. We configure it for using the smartcard:

```
[L0@Mac-mini ~]$ brew install openc
```

We try the authentication:

```
[L0@Mac-mini ~]$ ssh -I openc-pkcs11.so test-user@192.168.3.120  
Enter PIN for 'User PIN (OpenPGP card)':
```

```
Session recording is disabled.  
Audit logs recording is disabled.  
Session lock is disabled.  
Session's max duration is unlimited.
```

```
[test-user@centos7-client ~]$ pwd  
/home/test-user  
[test-user@centos7-client ~]$ exit  
exit
```

```
>>>> Session's duration was aprox 42 seconds <<<<
```

```
Connection to 192.168.3.120 closed.
```

I'm connected to the server with a user from the LDAP database and authenticated with my PIV key.

This manual was prepared with great care. However, RCDevs S.A. and the author cannot assume any legal or other liability for possible errors and their consequences. No responsibility is taken for the details contained in this manual. Subject to alternation without notice. RCDevs S.A. does not enter into any responsibility in this respect. The hardware and software described in this manual is provided on the basis of a license agreement. This manual is protected by copyright law. RCDevs S.A. reserves all rights, especially for translation into foreign languages. No part of this manual may be reproduced in any way (photocopies, microfilm or other methods) or transformed into machine-readable language without the prior written permission of RCDevs S.A. The latter especially applies for data processing systems. RCDevs S.A. also reserves all communication rights (lectures, radio and television). The hardware and software names mentioned in this manual are most often the registered trademarks of the respective manufacturers and as such are subject to the statutory regulations. Product and brand names are the property of RCDevs S.A. © 2021 RCDevs SA, All Rights Reserved