# DEV Guide

This document is intended to be updated on a regular basis as needed. Please be aware that the version you are currently looking at is dated: **October 5th, 2021**

Migrating from UDEV to DEV is a fairly straightforward process, but there are some new requirements and procedures waiting on the other side. This guide is geared towards helping you navigate these changes so you can get back to productivity as quickly as possible.

## Migrating Your Account

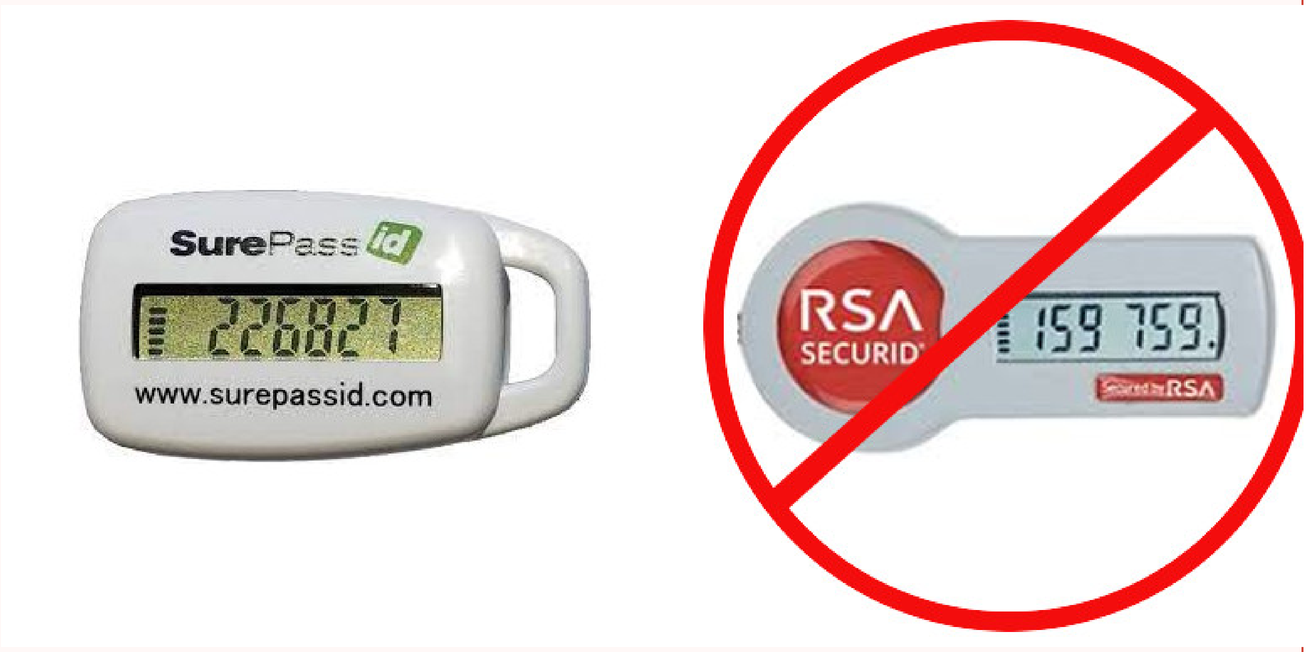
Though the term **migration** is being used it is better to think of this as **forking** your account into a new domain. Your account in the soon-to-be-deprecated UDEV domain will still be used to access all UDEV resources, including workstations, virtual machines, Atlassian systems, and Mattermost chat server. Updating information such as a password, SSH key, or group membership in one account will not be reflected in the other. All subsequent account updates will need to be manually mirrored where appropriate.

### Step 1: Put in a Helpdesk Ticket

Put in a ticket either by email (helpdesk@bia-boeing.com) or by going to https://apps.bia-boeing.com/helpdesk informing the IT Department that you wish to have your account migrated. The IT Department will then refresh your account on the new DEV domain.

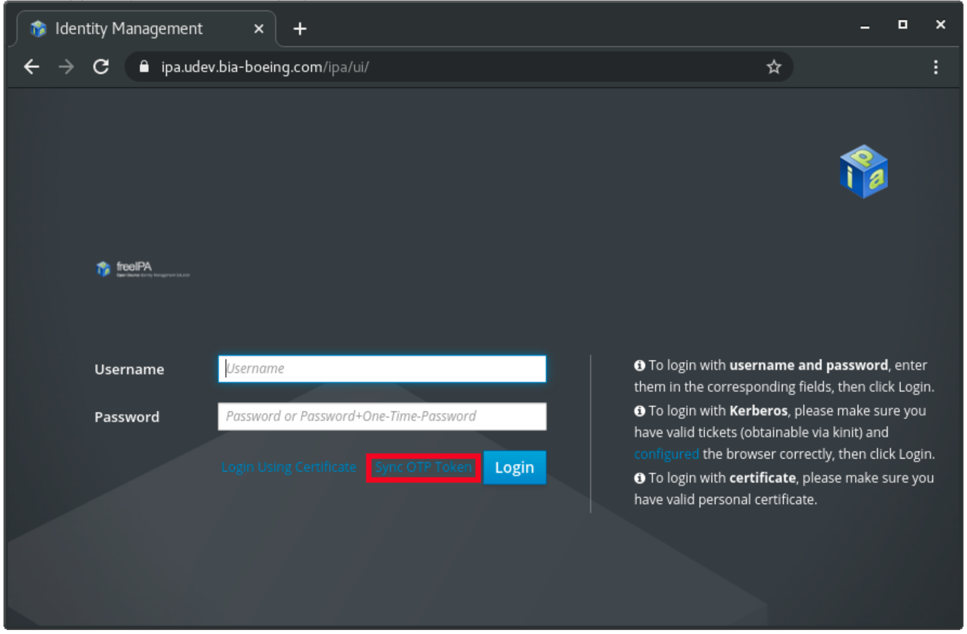
### Step 2: Sync Your SurePassID Token

You must use a white SurePassID token for MFA. This is not the same grey RSA token used to access company email, or customer resources.



1. Go to https://ipa-prime.dev.bia-boeing.com

2. Click on Sync OTP Token.  If you don't see this screen (i.e. you are already logged in), select "Logout" in the drop-down menu under your name in the upper right corner of the page.



Username: Username

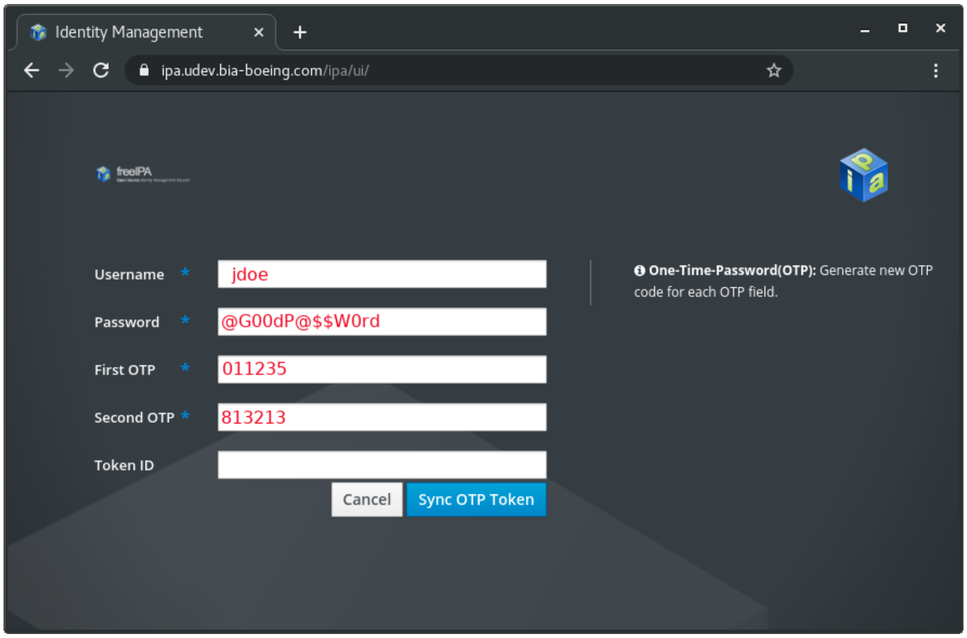
Password: Password or Password+One-Time-Password

ℹ To login with **username and password**, enter them in the corresponding fields, then click Login.

ℹ To login with **Kerberos**, please make sure you have valid tickets (obtainable via kinit) and configured the browser correctly, then click Login.

ℹ To login with **certificate**, please make sure you have valid personal certificate.

3. Fill out the fields and click Sync OTP Token
   a. Your password is your *First Factor* and should not include a code from your token
   b. First OTP is a code from your SurePassID token.
   c. Second OTP is the next code to appear on your SurePassID token. The codes **must** be sequential.
   d. Leave the Token ID field blank.



Username * jdoe

Password * @G00dP@$$W0rd

First OTP * 011235

Second OTP * 813213

Token ID

**One-Time-Password(OTP):** Generate new OTP code for each OTP field.

4. If successful you will be re-directed back to the home page and you should see a green dialog acknowledging your success.

## Step 3: Perform Migration

After syncing your token you will need to go to https://ipa-prime.dev.bia-boeing.com/ipa/migration and follow the prompts to activate your account in the new DEV domain.

Top of Page

# New Credential Requirements

In order to stay compliant with all of our policies and obligations credential requirements are getting tightened up. There are two key aspects to be aware of:

## Direct logins

Direct logins are anywhere you use a username and password. These are required to be MFA enabled with your SurePassID token. Passwords do not expire, though you are obligated to notify IT and to reset your password if you think it may have been compromised

Complexity requirements for passwords:

- Minimum 14 characters long
- At least 2 special characters
- At least 2 digits
- At least 2 lowercase letters
- At least 2 uppercase letters
- Cannot reuse the same password within 24 iterations

## SSH Authentication

### Private versus Public keys

An extremely important concept in Public Key Infrastructure (PKI) asymmetrical cryptography is the pairing of **private** and **public** keys.

- **Private Keys** allow you to decrypt messages sent to you that have been encrypted with their corresponding *public key*. It is paramount to the integrity of your keys to protect your private key and to never share it. If you ever need to move it to a different system you should make sure it is encrypted in transit, or else erase and revoke the old key and generate a new pair on the new system.
- **Public Keys** only allow someone/something that has them to encrypt messages in such a way that only holders of the private key can read them. Unlike private keys, public keys can safely be shared over unencrypted channels.

While not required, adding a password/passphrase will increase their security. Should you do so automated systems will not be able to use your key to authenticate as you without storing your password. Since this means that potentially several applications would need to cache your passphrase, it decreases the added protection. However adding a passphrase to a private key before moving it is a sound method of protecting it in transit.

### Key Types

No matter which way you create your key you can chose from a variety of types. However, given our need to remain FIPS compliant going forward it is recommended that you chose from either RSA or ECDSA.

- **RSA** is the most commonly type of ssh key used today. The default bit length of 2048 is sufficient in most cases, but upping to 4096 bits is encouraged.
- **ECDSA** is based on relatively newer cryptography, which allows it to offer brute force protection comparable to RSA at significantly smaller key sizes, and it is generally faster (though end users are unlikely to notice a difference). Unless you have reason to do otherwise, it is recommended to stick to the 256 bit length, though 384 and 521 are available.

It is recommended that any new keys be created using **ECDSA** algorithm with a bit length of **256**.

### Creating a New Key Pair

You can create a new key pair from the terminal with **ssh-keygen**.

```
[nmiller@07-l-8zz8zd3 .ssh]$ ssh-keygen -t ecdsa -f ./id_ecdsa
Generating public/private ecdsa key pair.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in ./id_ecdsa.
Your public key has been saved in ./id_ecdsa.pub.
The key fingerprint is:
SHA256:mgPSNb9JP8mj7ySoGh35fqLsKOuMk1TqK+zS3/LcDzg nmiller@07-l-8zz8zd3.dev.bia-boeing.com
The key's randomart image is:
+---[ECDSA 256]---+
|                 |
|                 |
|      o          |
|   o o o         |
|  + =   S        |
| o o + * = .     |
|+o. . E = B      |
|Oo.+.=.+.= o     |
|BO++*+=oo++      |
+----[SHA256]-----+
```

There are a few options worth looking at when creating a key pair:

- -t : Type, you can choose from RSA or ECDSA. The default is RSA, but this standard, while currently the most widely accepted, is being deprecated especially by FIPS compliant systems.
- -b: Bit Length, the larger the number the stronger the strength of the key. For RSA keys this must be at least 2048, though 4096 is recommended. Additionally, the length is an arbitrary amount, though powers of 2 are customary. For ECDSA key the lengths can be 256 (default), 384, or 521.
- -f: File, specify an output file. The default is to create a pair at ~/.ssh/id_<type> (private key) and ~/.ssh/id_<type>.pub (public key). When specifying a file you are only giving the private key's location, the public key will be generated automatically with an appended .pub.
- -C: Comment, you can add a custom comment to a key to help you identify it. The default is <username>@<fqdn>.

### Key Permissions and Labels

It is important that your keys have the correct permissions and SELinux context labels. Below demonstrates how to check, and what values you should see.

```
[nmiller@07-l-8zz8zd3 .ssh]$ ls -lZ
total 16
-rw-r-----. 1 nmiller nmiller unconfined_u:object_r:user_home_t:s0  222 Aug 17 14:04 config
-rw-------. 1 nmiller nmiller unconfined_u:object_r:user_home_t:s0  537 Aug 17 14:09 id_ecdsa
-rw-r-----. 1 nmiller nmiller unconfined_u:object_r:user_home_t:s0  201 Aug 17 14:09 id_ecdsa.pub
-rw-r-----. 1 nmiller nmiller unconfined_u:object_r:user_home_t:s0 2417 Aug 17 14:04 known_hosts
```

### Adding Your Public Key to Your DEV Profile

Log in to https://ipa-prime.dev.bia-boeing.com.

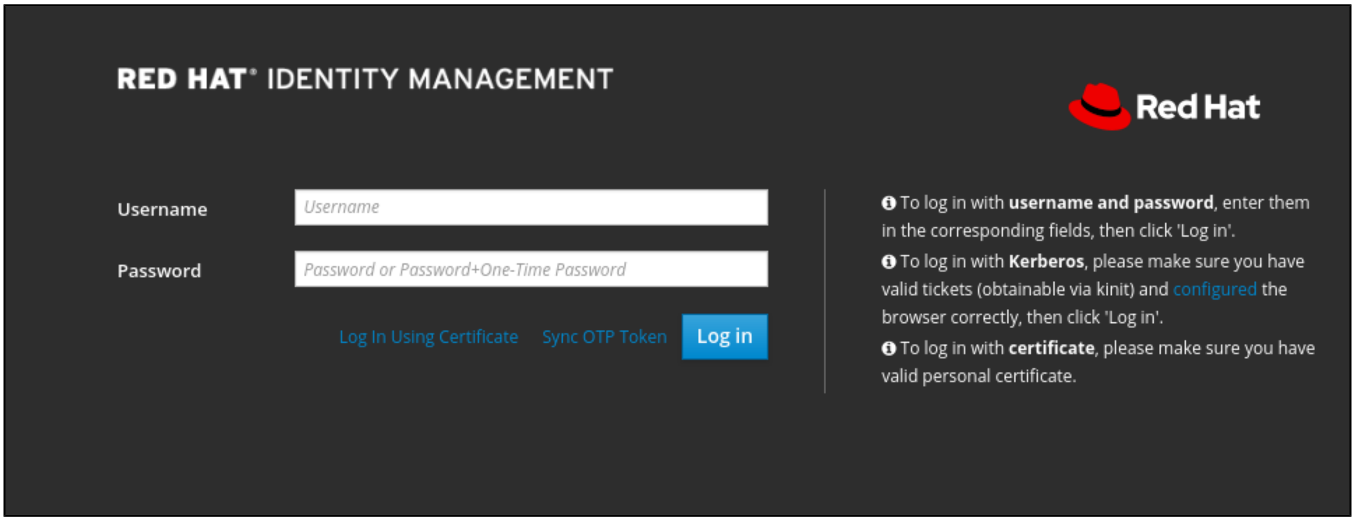
**TLS Warning**

If you receive a TLS warning when accessing https://ipa-prime.dev.bia-boeing.com this is expected. The server is using a self signed certificate at this time. You will need to **Accept the risks and continue** when prompted by your browser.

**Popups**

If your browser pops up login dialogs, close them out. This is a known issue for Chrome and Chromium based browsers (such as Edge). It is unfortunately not something we can control/prevent.

This the page you should see:

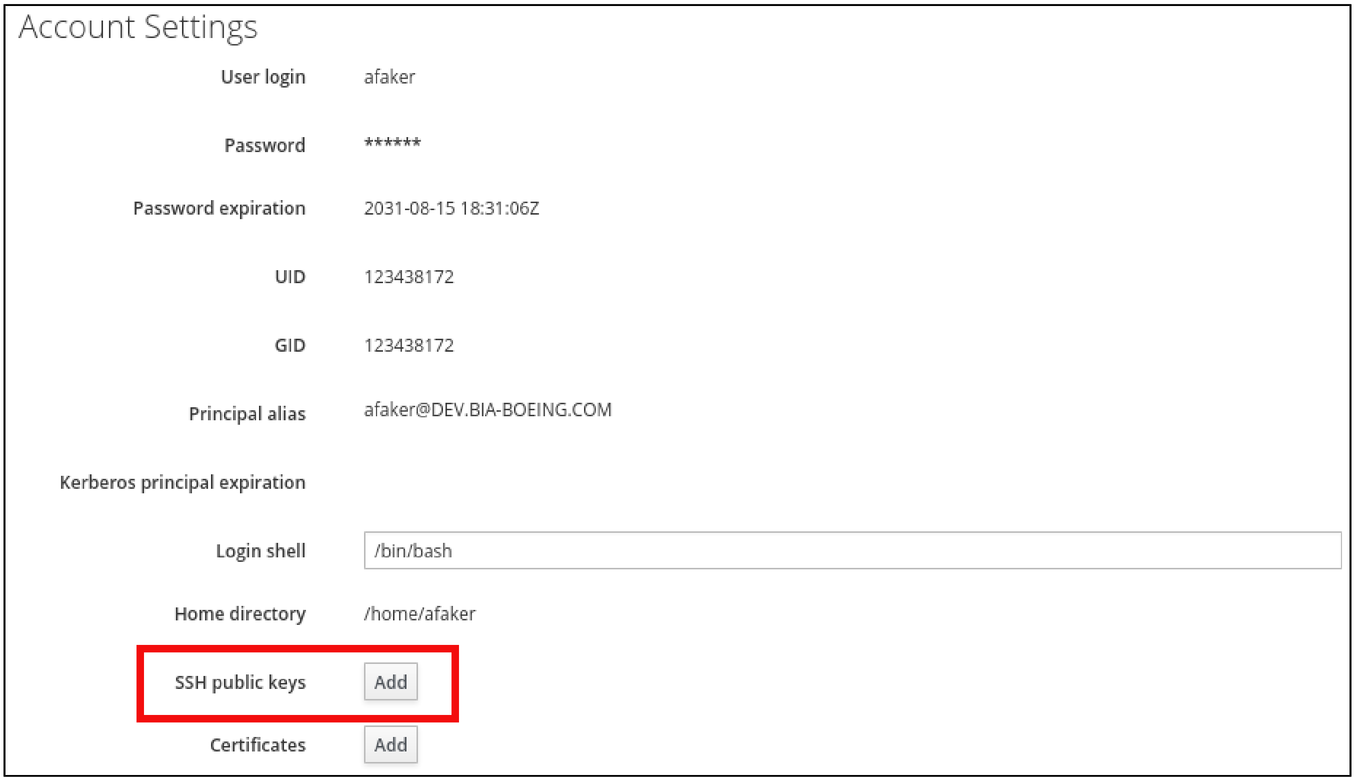


To log in with **username and password**, enter them in the corresponding fields, then click 'Log in'.

To log in with **Kerberos**, please make sure you have valid tickets (obtainable via kinit) and configured the browser correctly, then click 'Log in'.

To log in with **certificate**, please make sure you have valid personal certificate.

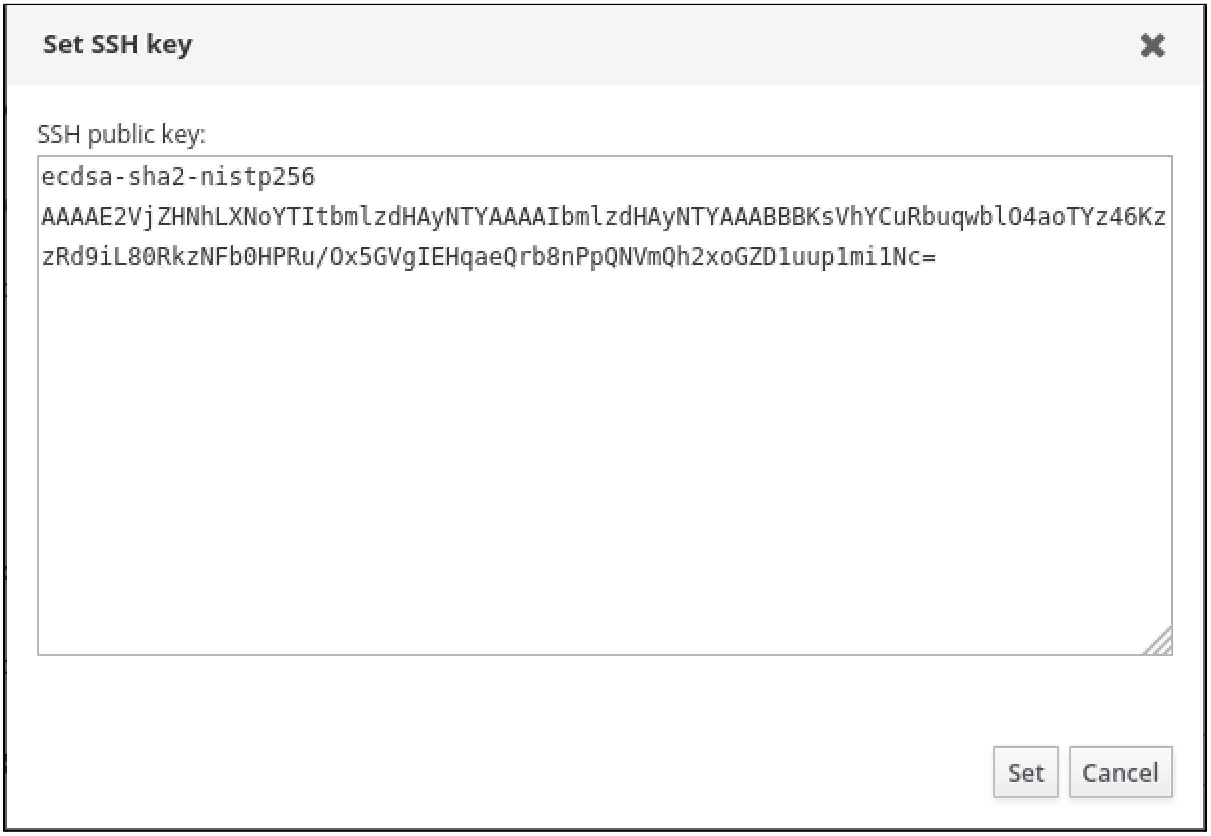
Once logged in, scroll down to **Account Settings**  **SSH public key**  and click **Add**



Account Settings

| | |
|---|---|
| User login | afaker |
| Password | ****** |
| Password expiration | 2031-08-15 18:31:06Z |
| UID | 123438172 |
| GID | 123438172 |
| Principal alias | afaker@DEV.BIA-BOEING.COM |
| Kerberos principal expiration | |
| Login shell | /bin/bash |
| Home directory | /home/afaker |
| SSH public keys | Add |
| Certificates | Add |

In the popup paste the contents of your public key:

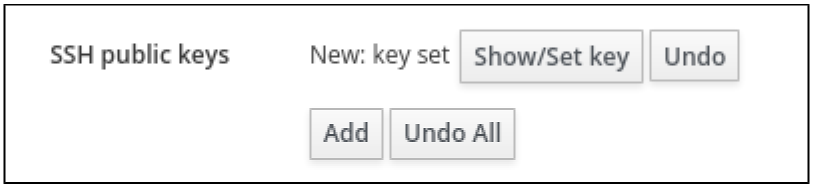


Set SSH key

SSH public key:

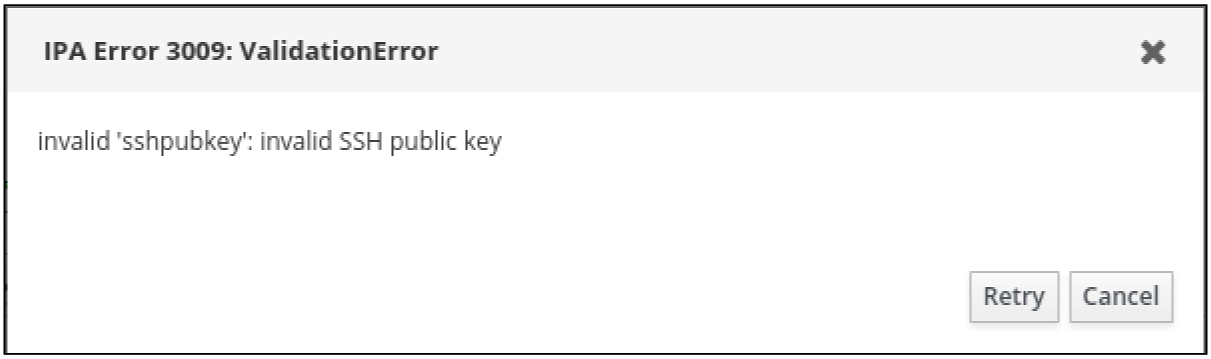
```
ecdsa-sha2-nistp256
AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHAyNTYAAABBBKsVhYCuRbuqwblO4aoTYz46Kz
zRd9iL80RkzNFb0HPRu/Ox5GVgIEHqaeQrb8nPpQNVmQh2xoGZD1uup1mi1Nc=
```

Click set and you should now see the following in your profile:



SSH public keys New: key set

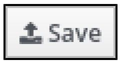
If your receive and error such as the one shown here you most likely copied the key incorrectly. This can happen quite easily be accidentially including a newline or space character, or by leaving off the prefix that asserts the key type/cipher.



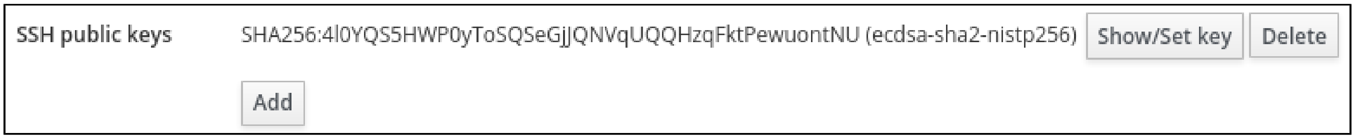
IPA Error 3009: ValidationError

invalid 'sshpubkey': invalid SSH public key

Once you have set a valid public key you must **Save** the changes to your profile. The icon towards to top of the profile looks like this:



You can confirm that the key was added to your profile successfully by scrolling back down and looking for a key fingerprint:



SSH public keys SHA256:4l0YQS5HWP0yToSQSeGjJQNVqUQQHzqFktPewuontNU (ecdsa-sha2-nistp256)

Top of Page

## Using the New Laptops

The new laptops, besides running RHEL 8, incorporate some new security features that necessitate a more involved start up procedure.

### Full Disk Encryption

The first thing you will notice is that when you boot up your laptop you will be required to type in a password to decrypt the hard drive, before the OS can even get started.

> ⓘ If you have not received your full disk encryption password, please contact IT.

### Zero Credential Caching

In the past credentials were able to be cached locally, enabling users to authenticate even when they could not connect to the Identity Management (IdM) server. Unfortunately, this also meant that bypassing MFA was as simple as blocking access to the IdM server, which trivially negated the security guarantees. Going forward all authentication must be done *online* with open communication to the IdM server. The following sections will explain how we are dealing with this complication.

### Connecting to the VPN

Just like before, you will need to connect to the VPN in order to authenticate. This is accomplished with the following procedure:

> ⓘ **Laptop Keyboard**
>
> If you are using the laptop's built in keyboard you will need to press the Function key (**fn**) when using the F keys.

1. Pressing **Ctrl + Alt + F2** to open a terminal session.
2. Log in with the same **vpn_access** account as before.
3. When prompted use your **first.last** username (as it appears in your @bia-boeing.com email address).
4. The password is your **PIN + RSA token code** (this is the token you use to access portal.office365.us, not the SurePassID token).
5. Press **Ctrl + Alt + F1** to go back to a GUI login screen and log in as usual.

> ⓘ **NumLock**
>
> By default the sessions will start with NumLock disabled. Enabling NumLock in one session will not enable it in another.

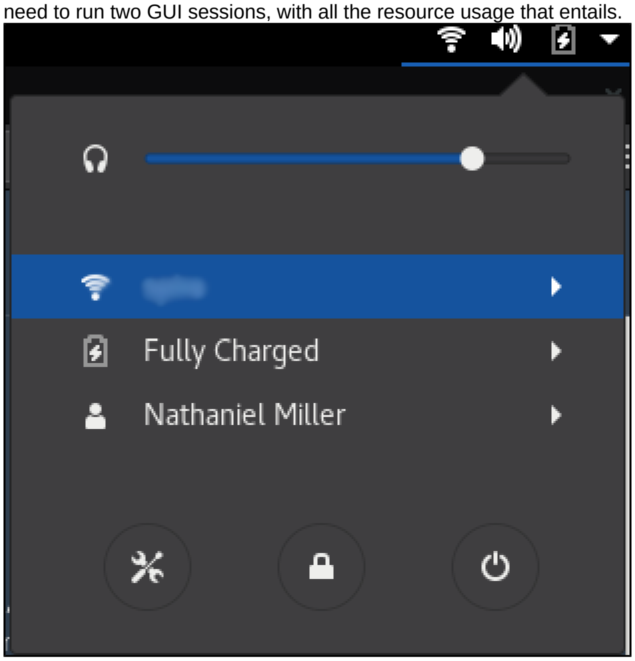
### Re-Connecting to the VPN

If the VPN connection should ever drop you will need to restart it. To do so you mostly follow the previous instructions, but with one twist. When you switch over to the terminal session you will be prompted to unlock the session with the **vpn_access** password. This will immediately kill the session if the VPN connection is down, and prompt you to log in again.

### First Time Network/WiFi Setup

Since there aren't any cached credentials you will not be able to login with your account to setup WiFi, or any other authenticated network connection. There are two options for proceeding:

- GUI: Login (from the GUI login screen) with the **vpn_access** account. Use the drop down menu from the top right corner (pictued below) to select and connect to a WiFi network. It is recommended that after setting up your connection you log out and use the standard connection methodology. Alternatively, you could establish a connection by opening a terminal in the GUI session, though this means your system will then need to run two GUI sessions, with all the resource usage that entails.



- NMCLI: You can also setup a WiFi network connection from the command line using nmcli. Here is a guide on how to do just that: https://www.makeuseof.com/connect-to-wifi-with-nmcli/

## GUI Login Screen

At the GUI login screen you will be prompted for your username. This is your DEV account, which in most cases is your *first initial and last name*. After hitting enter, you should get a prompt for your **First Factor**. This is your DEV password. Next you will be prompted for your **Second Factor**, which is the code supplied on your white SurePass ID token, which refreshes every 30 seconds.

> In you are prompted for your **Password** this is an indication that your system cannot reach the authentication service (Identity Management, ipa-prime.dev.bia-boeing.com). You should cancel your authentication attempt and try again. If you are still getting prompted only for your **Password** ensure that your system has an active VPN connection.

Top of Page

## Related articles

- DEV Guide
- Migrate Your Account to DEV
- SSH Keys for DEV