



A Recipe for Migrating to Go Modules

Modules are the Default

...and migrating is easier than
you think!





Mender.io

- ▶ Used for large-scale IoT deployments
- ▶ Build as a collection of microservices
- ▶ Varying version tags between services





Get a go.mod file

```
`go mod init`
```

v2+

- ◀ Glide
 - ◀ Godeps
 - ◀ Govendor
 - ◀ Dep
 - ◀ ... and more!
- ◀ Slightly more complex
 - ◀ Increment major version
 - ◀ Add to module path
 - ◀ Causes some other issues to address

```
`go mod init github.com/mendersoftware/deviceauth/v3`
```



Update imports

- ▶ Build and test
- ▶ Both will likely fail
- ▶ Update import statements for new path
- ▶ May want to use a script for this in large projects

```
- "github.com/mendersoftware/deviceauth/utils"  
+ "github.com/mendersoftware/deviceauth/v3/utils"
```



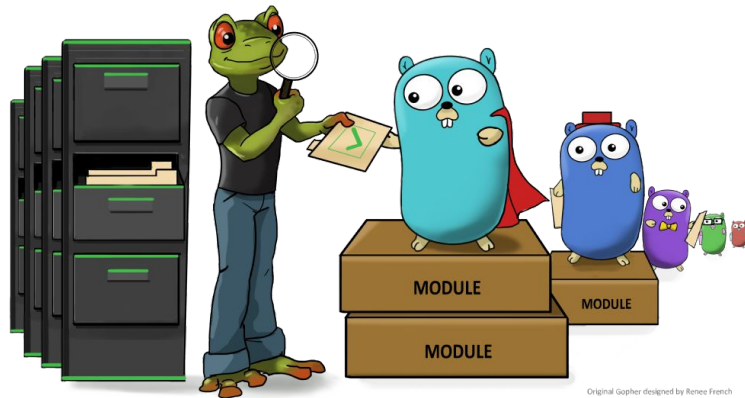
Module Versioning

- ▶ After ``go mod init`` or ``go mod tidy``, check version numbers
- ▶ Some dependencies may cause problems here
- ▶ Untagged changes in satori/go.uuid broke Mender
- ▶ At this point, you should be able to build and run tests

Module Proxy

- ▶ JFrog GoCenter
- ▶ Immutable and persistent modules
- ▶ Much faster than cloning
- ▶ Consistent, secure, reproducible builds

```
`export GOPROXY=https://gocenter.io`
```



Original Gopher designed by Renee French



Static Analysis and Code Generators

- ▶ Some projects may need to handle additional issues with paths here
- ▶ Remember to look at your .proto and Dockerfiles

Testing Changes

- ▶ Go v1.13 made changes to testing
- ▶ Specifically, when flags are parsed
- ▶ Move `flag.Parse()` to main test function

39	-	<code>flag.Parse()</code>
40	39	<code>}</code>
41	40	
42	41	<code>func TestRunMain(t *testing.T) {</code>
43	42	<code> if !runAcceptanceTests {</code>
44	43	<code> t.Skip()</code>
45	44	<code> }</code>
46	45	
46	+	<code>flag.Parse()</code>

Thanks!

Any questions?

You can find me at

- ✦ @Dixie3Flatline
- ✦ katc@jfrog.com
- ✦ jfrog.com/shownotes

