

# Keine Angst vor Puppet 4

Secure Linux Administration Conference 2015

Martin Alfke

<ma@example42.com>

# Martin Alfke

- ✦ PL Training Partner
- ✦ Module Contributor
  
- ✦ Freelancer / example42
- ✦ Infrastructure Architect



# Workshop

- VMs:
  - Puppet 3:
    - <http://10.60.3.179/SLAC> Puppet Workshop.ova
  - Puppet 4:
    - <http://10.60.3.179/SLAC> Puppet4 Master.ova

# Puppet 4

- ✦ 15. April 2015
- ✦ <https://puppetlabs.com/blog/say-hello-open-source-puppet-4>
- ✦ <http://docs.puppetlabs.com/puppet/4.0/reference/index.html>





# Deprecations



# Node Inheritance

```
node 'basenode' {                                     # Ein Dummy Node, der als default
  include base                                       verwendet wird
  include security
}
```

```
node 'www.server.com' inherits basenode {           # Der echte Node wird vom Dummy
  include webserver                                 Node "vererbt"
}
```

# Node Inheritance

```
node 'www.server.com' {  
  include profile::webserver  
}
```

# Node Klassifizierung ohne Vererbung

```
class role::basenode {  
  include role::base  
  include role::security  
}
```

# Roles & Profiles - Klassen definieren  
und deklarieren

```
class profile::webserver {  
  include role::basenode  
}
```

# Vergleich von “

```
$message = “
```

Leerer String als default

```
if $message {  
  notify { “Message: ${message}”: }  
}
```

Check Existenz der Variable mit Inhalt

```
$message = “
```

Leerer String als default

```
if $message and $message != “ {  
  notify { “Message: ${message}”: }  
}
```

Check Existenz der Variable mit nicht leerem Inhalt



# Namen von Variablen

- ✦ Eine Variable darf nicht
  - ✦ mit einem Großbuchstaben anfangen
  - ✦ mit einem Unterstrich anfangen ... (naja, sie darf. Aber. Dann ist es eine private Variable.)

# Namen und Syntax von Referenzen

- ✦ Reference deprecation
  - ✦ Title mit Großbuchstabe
  - ✦ Leerzeichen zwischen Type Referenz und Title

Class [Ssh]

Titel mit Großbuchstabe

Class ['ssh']

Leerzeichen

Class['ssh']

OK

# Minus Zeichen

- ✦ Keine Minuszeichen in
  - ✦ Modul Namen
  - ✦ Klassen Namen
  - ✦ Define Namen

# Noch mehr

- ✦ Relativer Namensraum von Klassen - der Grund warum man den doppelten Doppelpunkt bei include verwenden musste - `include ::ssh`
- ✦ Importieren von Manifesten - `import '*.pp'`
- ✦ Vergleiche von unterschiedlichen Typen
  - ✦ Nummern mit Regulären Ausdrücken
  - ✦ Bool mit String



# Neu in Puppet 4



RICK BAKER  
© 07



# EPP Templates

- ✦ Puppet \$var anstelle von Ruby @var
- ✦ `epp(filename)`
- ✦ `inline_epp(epp_string)`

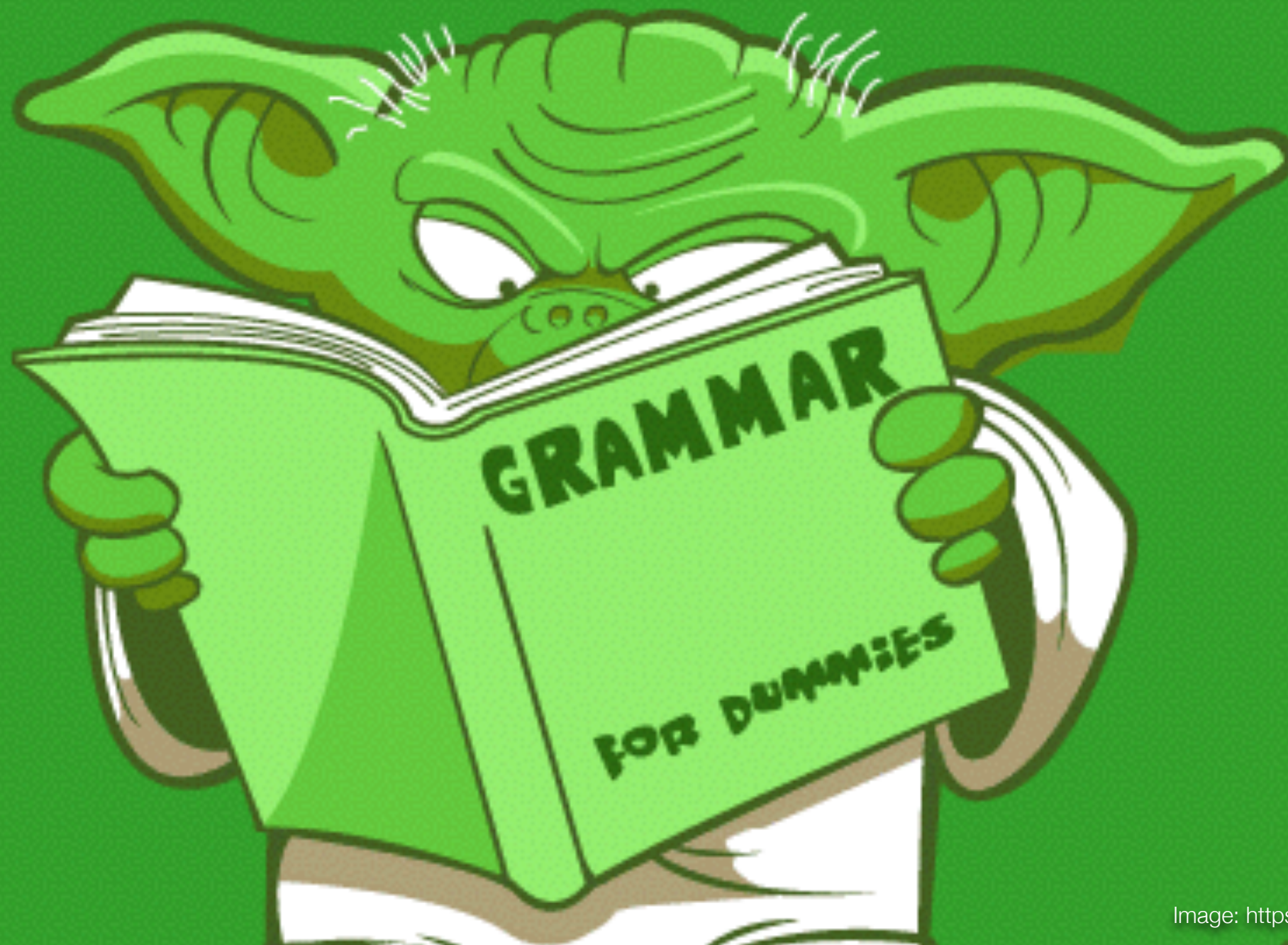
# Lambdas und Funktionen

- ✦ each - über ein Array iterieren
- ✦ map - Array oder Has in neues Array umwandeln
- ✦ filter - Arrays oder Hashes filtern
- ✦ reduce - Array oder hash auf einzelnen Wert reduzieren
- ✦ slice - Array oder Hash in Slices schneiden

# Verwendung von Funktionen

- ✦ Standard Puppet Syntax:
  - ✦ `function_name(argument) - each($variable)`
- ✦ Ruby Syntax:
  - ✦ `argument.function_name - $variable.each`

# Types, Types, Types



# Datentypen

- Integer[from, to]
- Float[from,to]
- Enum[\*strings]
- Pattern[\*patterns]
- Regexp[regexp]
- Boolean
- Array
- Hash



# Warum Daten Typen?

```
class ssh (  
  $server = true,  
) {  
  if $server {  
    include ssh::server  
  }  
}
```

```
class { 'ssh':  
  server => 'false',  
}
```

# Warum Daten Typen?

```
class ssh (  
  $server = true,  
) {  
  if validate_bool($server) {  
    include ssh::server  
  }  
}
```

```
class { 'ssh':  
  server => 'false',  
}
```

# Deklaration von Datentypen

```
class ssh (  
  Boolean $server = true,  
) {  
  if $server {  
    include ssh::server  
  }  
}  
•
```

# Deklaration von Datentypen

```
class ssh (  
  Boolean $server = true,  
) {  
  if $server {  
    include ssh::server  
  }  
}
```

```
class { 'ssh':  
  server => 'false',  
}
```

Error 400 on SERVER: Expected parameter 'server' of 'Class[Ssh]' to have type Boolean, got String

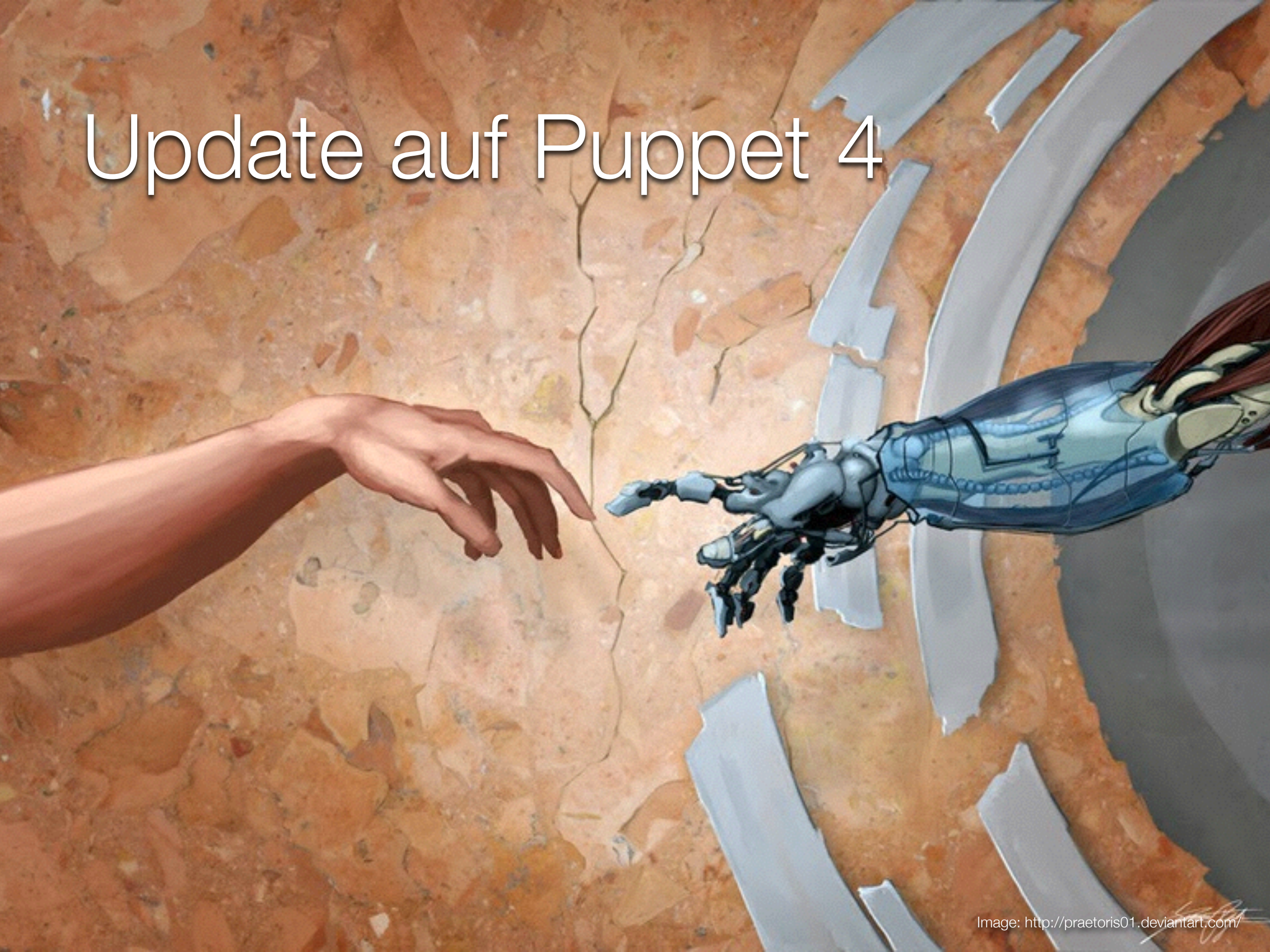
# Komplexe Datentypen

```
class users (  
  Hash $hash  
) {  
  $userarray = keys($hash)  
  users::user_data { $userarray: }  
}
```

```
define users::user_data (  
  String   $gid           = $users::hash[$title][gid],  
  String   $home          = $users::hash[$title][home],  
  Boolean $managehome    = $users::hash[$title][managehome],  
) {  
}
```



# Update auf Puppet 4





# Ruby Hölle

- Puppet & Ruby Versionen
  - puppet 3 - ruby 1.8
  - puppet 4 - ruby 1.9.3
- Lösung:
  - ruby 1.9.3 Pakete - wenn vorhanden, CentOS?
  - Puppet 4 - bringt ruby mit

# Puppet 4 Master

- Aufsetzen Puppet 4 Master
- Umziehen der SSL Zertifikate
- Puppet Konfiguration:
  - `ca_server`: auf Puppet 3 Master & Agents angeben
  - `server`: auf Puppet 3 Agents den alten Master angeben

# Pfade bei Puppet 4

- /opt/puppetlabs/bin
  - Wird nicht in PATH aufgenommen!
- /etc/puppetlabs/puppet/
  - ssl im Puppet Verzeichnis
- /etc/puppetlabs/puppetserver/
- /etc/puppetlabs/code
  - environments, hiera Daten

# Keine Angst vor Puppet 4

Secure Linux Administration Conference 2015

Martin Alfke

<ma@example42.com>