

Where Bacula meets
OpenBSD and friends

-25%

Agenda

- Who we are
- Why did we choose Bacula
- Why OpenBSD?
- Bacula on OpenBSD
- System Management with Puppet
- System monitoring
- Challenges

Who we are?

Open Source developers

Working at M:tier Ltd.

System development and
solution provider

Mostly blue chip customers



Why did we choose Bacula?

Enterprise ready

Open Source

Supports many platforms

Easily extendable



OpenBSD

Only two remote holes in the default install in more than 16 years

Open Source Unix like operating system

Secure design of the core system

Used where security matters

POSIX, portability and correctness



Why we use OpenBSD?

M:tier employees are mostly OpenBSD developers

Easy to deploy and install

Good documentation

Reliability. Stable interfaces.

We know it inside out



How we use Bacula on OpenBSD

Backup to Disk

Backup to Tape

Stable for 5 years



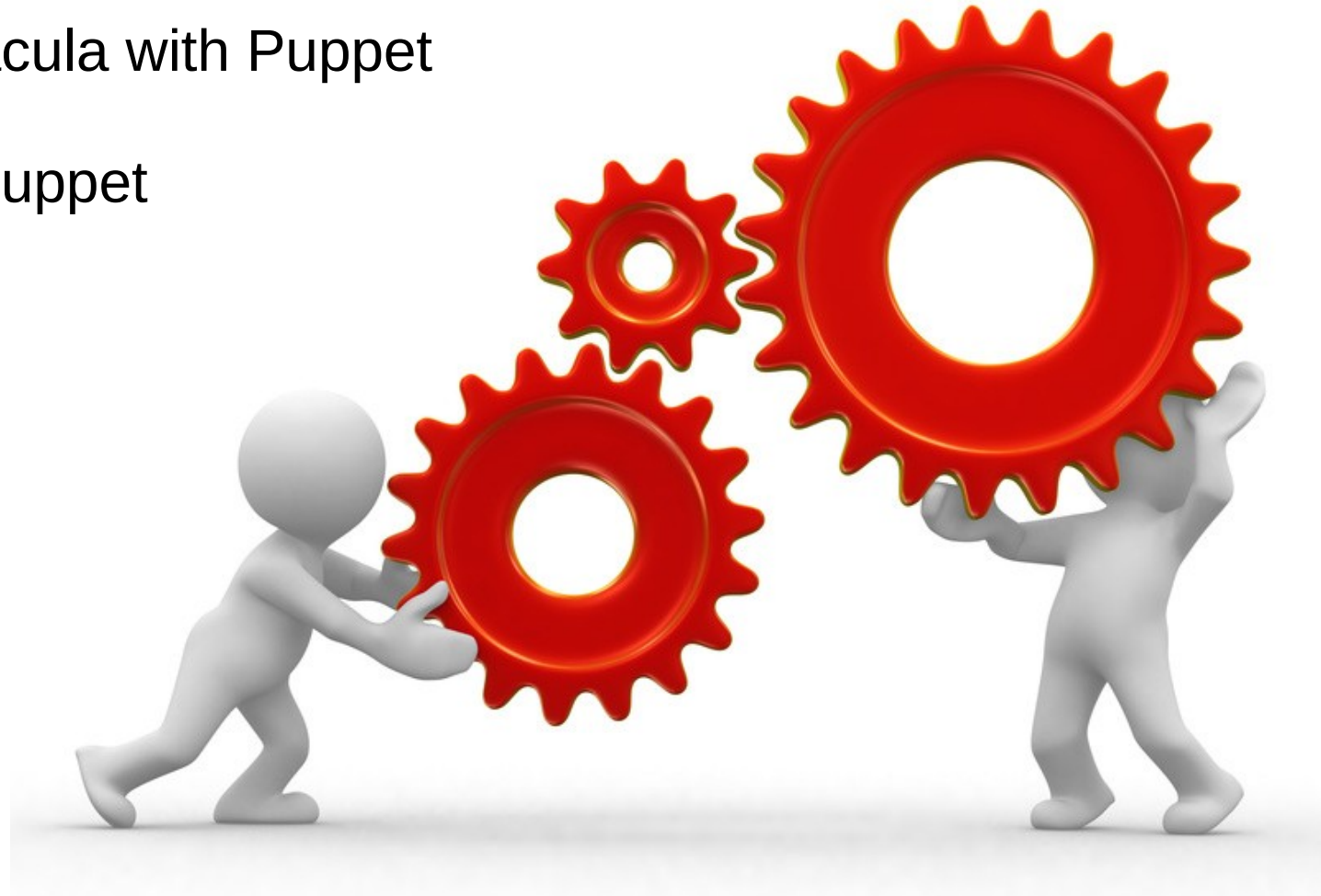
Puppet System Management

What is Puppet

How do we use Bacula with Puppet

Benefits of using Puppet

Puppet Dashboard



Puppet example configuration



Puppet bacula.dir extract #1

```
class bacula_dir inherits conf {
    require postgresql

    install_pkg { ['bacula-server--pgsql']: }
    # needed to handle dependencies
    install_pkg { ['bacula-server']: }

    file {
        "/etc/bacula/bacula-dir.conf":
            owner => '_bacula',
            group => '_bacula',
            mode => 0640,
            require => Package["bacula-server"],
            content => multi_source_template("/etc/bacula/bacula-dir.conf");
        "/etc/bacula/bacula-dir.conf.local":
            owner => '_bacula',
            group => '_bacula',
            mode => 0640,
            require => Package["bacula-server"],
            content => multi_source_template("/etc/bacula/bacula-dir.conf.local");
    }
}
```

Puppet bacula.dir extract #2

```
if ($::bootstrap) {
  exec {
    "create_bacula_db_user":
      command => "createuser -U postgres --no-superuser /
--createdb --no-createrole bacula",
      user => "root",
      path => "/bin:/sbin:/usr/sbin:/usr/bin:/usr/local/bin",
      require => Package["bacula-server"];
    "set_bacula_db_user_pass":
      command => "echo \"ALTER USER bacula with PASSWORD
'$bacula_db_pass';\" | psql -U postgres",
      user => "root",
      path => "/bin:/sbin:/usr/sbin:/usr/bin:/usr/local/bin",
      require => Exec["create_bacula_db_user"];
    "create_bacula_db":
      command => "/usr/local/libexec/bacula/create_bacula_database /
-U bacula",
      user => "root",
      path => "/bin:/sbin:/usr/sbin:/usr/bin:/usr/local/bin",
      require => Exec["set_bacula_db_user_pass"];
  }
}
```

Puppet bacula.fd extract

```
class bacula_fd inherits conf {  
  
    install_pkg { ['bacula-client']: }  
  
    file {  
        "/etc/bacula/bacula-fd.conf":  
            owner => '_bacula',  
            group => '_bacula',  
            mode => 0640,  
            require => Package["bacula-client"],  
            content => multi_source_template("/etc/bacula/bacula-fd.conf");  
        "/etc/bacula/bconsole.conf":  
            owner => '_bacula',  
            group => '_bacula',  
            mode => 0640,  
            require => Package["bacula-client"],  
            content => multi_source_template("/etc/bacula/bconsole.conf.restricted");  
    }  
}
```

Puppet bacula.sd extract

```
class bacula_sd inherits conf {
  require bacula_dir

  file {
    "%.b":
      ensure => directory,
      owner => '_bacula-sd',
      group => 'wheel',
      mode => 0700;
    "/etc/bacula/bacula-sd.conf":
      owner => '_bacula',
      group => '_bacula',
      mode => 0640,
      content => multi_source_template("/etc/bacula/bacula-sd.conf");
    "/etc/bacula/bacula-sd.conf.local":
      owner => '_bacula',
      group => '_bacula',
      mode => 0640,
```

bacula-sd.conf variables

```
Director {  
  Name = bacula-dir  
  Password = "<%= scope.lookupvar('conf::bacula_fd_pass') %>"  
}
```

```
Director {  
  Name = bacula-mon  
  Password = "<%= scope.lookupvar('conf::bacula_mon_pass')  
%>"  
  Monitor = yes  
}
```

```
FileDaemon {  
  Name = <%= scope.lookupvar('::hostname') %>-fd  
  FDport = 9102  
  WorkingDirectory = /var/bacula  
  Pid Directory = /var/run  
  Maximum Concurrent Jobs = 5  
}
```

Deployment and management time

-40%

Zabbix monitoring

What is Zabbix

What do we monitor

Auto restart services

ZBX-Mobile for Android and
iPhone



System availability

+40%

Challenges

Ramp up time / Initial costs

Very high learning curve for Bacula and friends

Killer GUI frontend for Bacula

Error messages are not always informative

Try and error on non-Linux platforms

Summary

- Secure solution
- Fast deployment
- Easy to manage
- Easy to monitor
- Where costs are reduced



Contact M:tier

<http://www.mtier.org>

contact@mtier.org