Configuration Management with Puppet.

Nate Childers
CM is a big topic, we’re focused on what’s typically called SCM. Configuration inventory (documentation), controlled change (change management), elimination of entropy.

Transparent Operations.

Why do we care? The answer is simple: Take what we do and make it repeatable.
Automation.

Reliable. as good as we are we will always make mistakes.

rsync, tar, rm, useradd, the tools we use. we all know all the options and run them the same way each time right?
example - user creation

create user
set password?

this is easy right? we do this all the time.
create user
set password
populate content
install packages?
home directory?
scripts?
setup sudo
loosen firewall

Little bit more real world example here
what if we don’t want them to stick around forever?
why puppet?

Ruby, actively developed, commercial support available.
class chugalug::doit {
  user ("chugalug"):  
    ensure => present,
    password => "$yourcryptstringhere!",
    comment => "Classic Hackers UGA Linux Users Group",
  }

  file ("/home/chugalug"):  
    source => "/nfs/swamp/c/ch/chugalug",
  }

  package ("vim-enhanced"):  
    ensure => latest,
  }

  iptables { "let chugalug in because they friendly."
    ensure => present,
    port   => "tcp:22",
    source => "192.168.50.28/32",
    jump   => "ACCEPT",
  }
}
how to undo it in puppet code.

class chugalug::undoit {
  user {"chugalug":
    ensure => absent,
  }{
    file {"/home/chugalug":
      ensure => absent,
    }{
      package {"vim-enhanced":
        ensure => absent,
      }{
        iptables {"let chugalug in because they friendly.":
          ensure => absent,
        }{
          #
        }{
        }
      }{
    }
  }{
}
that was easy, right?

doesn’t have to be complicated, start simple and grow.
Adopting configuration management can improve your life.