

AUTOMATION
BOOTCAMP!
cPanel Conference '10

OCTOBER 4TH-6TH, 2010

Pluggable Authentication Framework

cPanel

Welcome

Pluggable Authentication

AUTOMATION
BOOTCAMP!
cPanel Conference '10

OCTOBER 4TH-6TH, 2010



John Lightsey

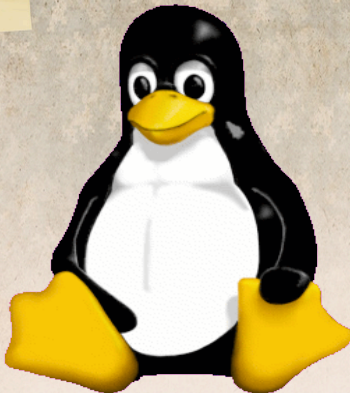
cPanel

My name is John Lightsey

AUTOMATION
BOOTCAMP!
cPanel Conference '10

Pluggable Authentication

OCTOBER 4TH-6TH, 2010



2000

cPanel

The image is a promotional graphic for a conference. It features a central illustration of Tux, the Linux mascot penguin, sitting on a light-colored, textured surface. Above the penguin, there is a yellow banner with the text 'AUTOMATION BOOTCAMP!' and 'cPanel Conference '10'. To the right, an orange banner displays the dates 'OCTOBER 4TH-6TH, 2010'. The top right corner contains the text 'Pluggable Authentication'. Below the penguin, the year '2000' is written in a large, black font. The bottom left corner features the 'cPanel' logo in orange and white. The entire graphic is set against a dark grey background.

Linux sysadmin since 2000

AUTOMATION

BOOTCAMP!

cPanel Conference '10

Pluggable Authentication

OCTOBER 4TH-6TH, 2010

`#!/usr/bin/perl`

2000

cPanel

Programming perl for as long

AUTOMATION

BOOTCAMP!

cPanel Conference '10

Pluggable Authentication

OCTOBER 4TH-6TH, 2010



Debian Swirl Logo is Copyright (c) 1999
Software in the Public Interest

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

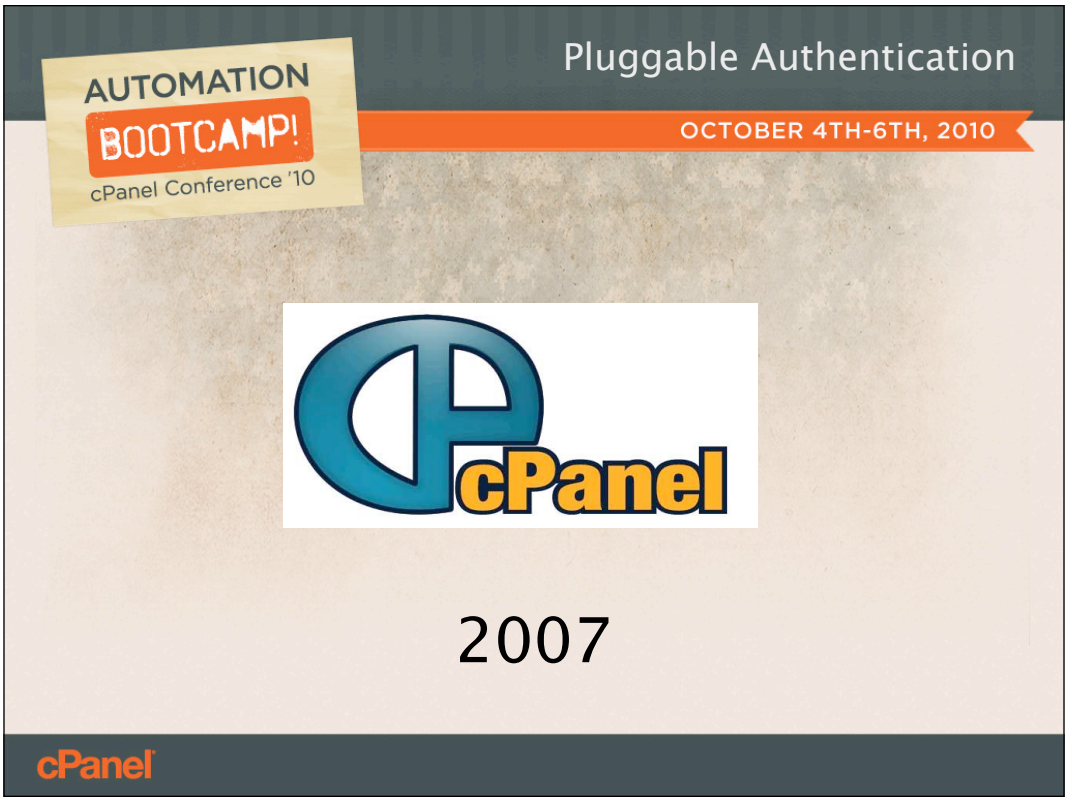
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

2005

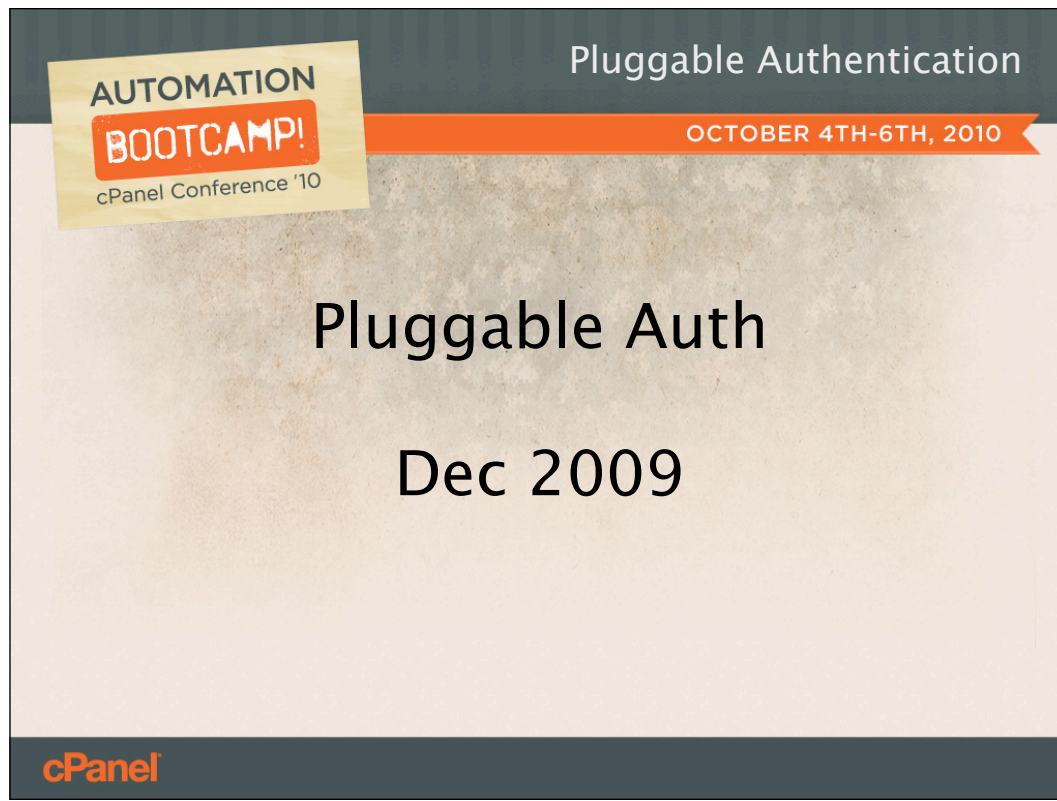
cPanel

Debian developer since 2005

NO WARRANTY



Cpanel Developer since 2007



Lead developer on the pluggable auth project since December of last year.

AUTOMATION
BOOTCAMP!
cPanel Conference '10

Pluggable Authentication

OCTOBER 4TH-6TH, 2010



</me>

cPanel

Thats me in a nutshell

AUTOMATION

BOOTCAMP!

cPanel Conference '10

Pluggable Authentication

OCTOBER 4TH-6TH, 2010

Disclaimers

cPanel

A few disclaimers before we begin

AUTOMATION

BOOTCAMP!

cPanel Conference '10

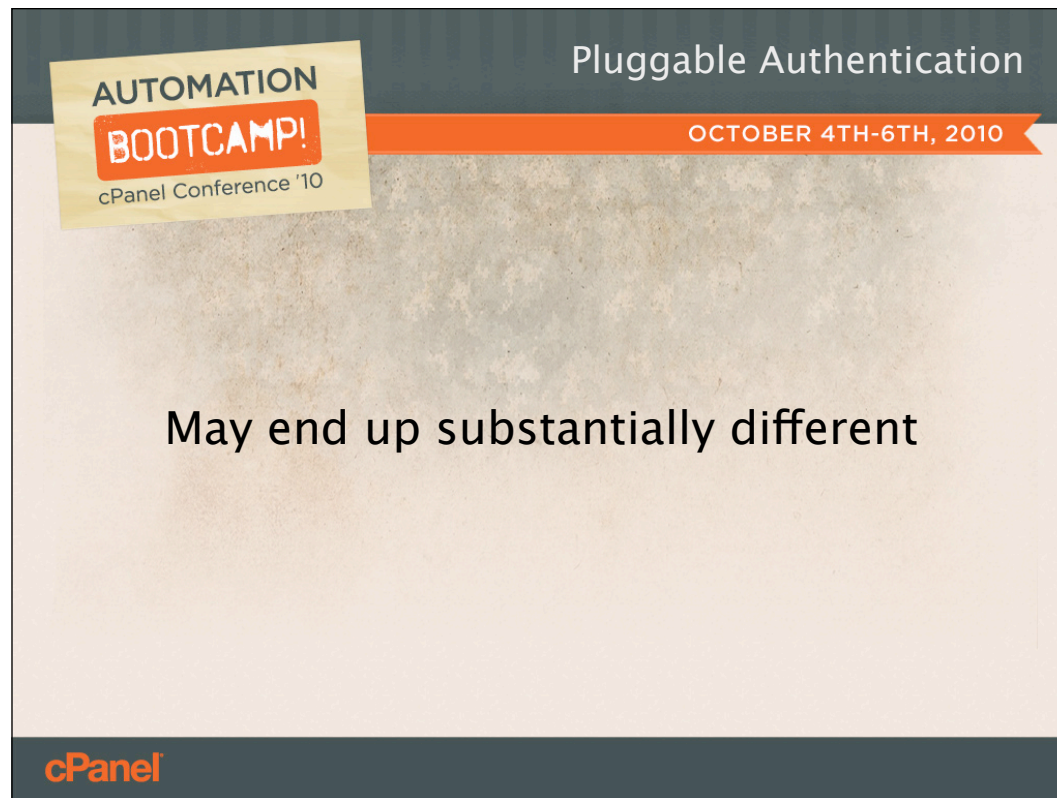
Pluggable Authentication

OCTOBER 4TH-6TH, 2010

Future functionality

cPanel

This talk is about proposed functionality for cPanel systems.



The pluggable auth framework is not in its final form and may end up looking substantially different.

AUTOMATION
BOOTCAMP!
cPanel Conference '10

Pluggable Authentication

OCTOBER 4TH-6TH, 2010

May not be complete during the scheduled
timeframe

cPanel

We have put out estimates of what release will include these changes, but those timetables may change in the future.



Overview of the talk

**AUTOMATION
BOOTCAMP!**
cPanel Conference '10

Pluggable Authentication

OCTOBER 4TH-6TH, 2010

A username and password are being requested by <https://jd.dev.cpanel.net:2083>. The site says: "cPanel"

User Name:

Password:

cPanel

Start with a broad description of how authentication currently works on cPanel systems.

**AUTOMATION
BOOTCAMP!**
cPanel Conference '10

Pluggable Authentication

OCTOBER 4TH-6TH, 2010

```
graph LR; User[User] --> PAF((Pluggable Auth Framework)); PAF --> Server[Server];
```

The diagram illustrates the Pluggable Authentication Framework. It features a central cloud-shaped box labeled "Pluggable Auth Framework". To the left of this cloud is a green rectangular box labeled "User", with an arrow pointing from the "User" box to the cloud. To the right of the cloud is a yellow rectangular box labeled "Server", with an arrow pointing from the cloud to the "Server" box. The entire diagram is set against a light brown background.

cPanel

Then we'll talk about the design of the new Pluggable Authentication Framework

AUTOMATION
BOOTCAMP!
cPanel Conference '10

Pluggable Authentication

OCTOBER 4TH-6TH, 2010



cPanel

Next we'll talk about customizing the new system.

AUTOMATION
BOOTCAMP!
cPanel Conference '10

Pluggable Authentication

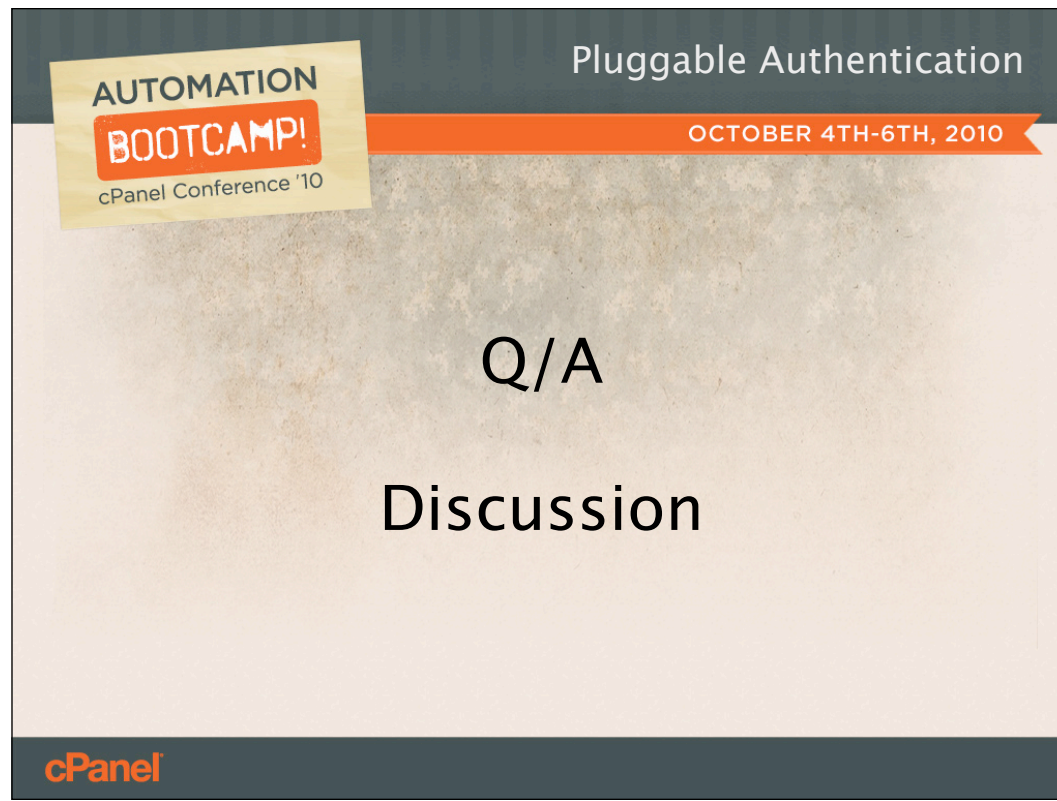
OCTOBER 4TH-6TH, 2010



More cowbell

cPanel

We'll discuss other improvements to authentication that have been rolled in with this project.



We'll conclude with a discussion session where you can voice any comments or questions you have about this new system.

Please hold any questions you may have until the end of the talk.



Jumping in, we'll start with a broad look at how authentication is currently handled on cPanel systems.



Authentication is all about gaining access to an account and accounts come in many different types.

AUTOMATION

BOOTCAMP!

cPanel Conference '10

Pluggable Authentication

OCTOBER 4TH-6TH, 2010

2 Primary Types

cPanel

There are two primary account types that most services handle.



2 Primary Types

- System

System accounts have a distinct UID and an entry in the `/etc/passwd` file.



2 Primary Types

- System
- Virtual

Virtual accounts share UIDs on the system and no entry in the `/etc/passwd` file.

AUTOMATION
BOOTCAMP!
cPanel Conference '10

Pluggable Authentication

OCTOBER 4TH-6TH, 2010

2 Secondary Types

cPanel

The slide features a dark grey header with the title 'Pluggable Authentication' in white. Below the header is an orange banner with the dates 'OCTOBER 4TH-6TH, 2010'. In the top left corner, there is a yellow sticky note graphic with the text 'AUTOMATION BOOTCAMP! cPanel Conference '10'. The main content area has a light beige background with the heading '2 Secondary Types' centered. The cPanel logo is in the bottom left corner of the slide.

Cpanel systems have two secondary types of accounts that go beyond what you'd find on a non-cpanel system.



2 Secondary Types

- cPanel

The first type is cPanel accounts. These are system accounts with extra metadata. The defining feature of a cpanel account is the cpusers file at `/var/cpanel/users/username`



2 Secondary Types

- cPanel
- WHM

WHM accounts are another type of secondary account and like cPanel accounts they correspond to system accounts.

AUTOMATION
BOOTCAMP!
cPanel Conference '10

Pluggable Authentication

OCTOBER 4TH-6TH, 2010

WHM Accounts

- Normal Resellers
- Hand Configured Resellers
- Root

cPanel

There are several different types of WHM accounts.

The most common are normal resellers which are cPanel accounts that have been given reseller privileges.

You can also take any system account and manually turn it into a reseller by adding it to the `/var/cpanel/resellers` file whether or not it has cPanel metadata.

Root is also special since it has no entry in the `/var/cpanel/resellers` file.

AUTOMATION
BOOTCAMP!
cPanel Conference '10

Pluggable Authentication

OCTOBER 4TH-6TH, 2010

The diagram consists of two concentric ovals. The outer oval is light blue and labeled "System Accounts". The inner oval is lime green and labeled "Secondary Accounts". This visualizes that secondary accounts are a subset of system accounts.

cPanel

So in a broad sense we have system accounts being a superset of the secondary accounts and those are further broken into subsets.

AUTOMATION
BOOTCAMP!
cPanel Conference '10

Pluggable Authentication

OCTOBER 4TH-6TH, 2010

Most services support virtual accounts

cPanel

The slide features a dark grey header with the title 'Pluggable Authentication' in white. Below the header is an orange banner with the dates 'OCTOBER 4TH-6TH, 2010'. In the top left corner, there is a yellow sticky note graphic with the text 'AUTOMATION BOOTCAMP! cPanel Conference '10'. The main body of the slide is a light beige color with a subtle texture, containing the text 'Most services support virtual accounts' in a black sans-serif font. The bottom of the slide is a dark grey footer with the 'cPanel' logo in orange.

Virtual accounts are mostly a feature of different services.

Pluggable Authentication

OCTOBER 4TH-6TH, 2010

AUTOMATION
BOOTCAMP!
 cPanel Conference '10

Service	System	cPanel	WHM	Virtual
SSH/SFTP	Yes	Yes	Yes	No
FTP	Yes	Yes	Yes	Yes
IMAP/POP/SMTP	Yes	Yes	Yes	Yes
Webmail	No	Yes	Yes	Yes
cPanel	No	Yes	Yes*	No
WHM	No	No	Yes	No
WebDAV	Yes	Yes	Yes	Yes
MySQL	No	No	No	Yes
Frontpage	No	No	No	Yes
Tomcat	No	No	No	Yes
Telnet	Yes	Yes	Yes	No

cPanel

Matrix of common services and the account types they support.

Mail services vs webmail

Cpanel login for WHM resellers

MySQL and Frontpage only support virtual accounts.

Turn telnet off



Lets take a quick look at the overall design of the present authentication system.

The present system follows one of four different design patterns depending on the individual service.

**AUTOMATION
BOOTCAMP!**
cPanel Conference '10

Pluggable Authentication

OCTOBER 4TH-6TH, 2010

```
graph LR; Service[Service] -.->|PAM| cPanelCode[cPanel Code]
```

The diagram illustrates a minimal integration pattern where a Service (represented by a green box) interacts with a cPanel Code (represented by a light blue box) through the Pluggable Authentication Module (PAM). A dashed arrow labeled 'PAM' points from the Service to the cPanel Code.

cPanel

The first pattern is where we have minimal integration with the authenticating service.

Generally we can read and write the password files to manage accounts but do little beyond that.

In some circumstances we use PAM to provide limited access controls.



Minimal integration:

- SSH/SFTP
- ProFTPD
- MySQL
- Frontpage

Most of the services following this pattern have poor integration with cPanel's auth systems.

Examples:
SSH
ProFTPD
MySQL
Frontpage



On the opposite end we have cPanel provided services where we control everything from end to end.

AUTOMATION
BOOTCAMP!
cPanel Conference '10

Pluggable Authentication

OCTOBER 4TH-6TH, 2010

cPanel provided:

- Webmail
- cPanel
- WHM
- WebDAV

cPanel

Examples of cPanel provided services

Cpsrvd:

Webmail
Cpanel
WHM

Standalone:
WebDAV

**AUTOMATION
BOOTCAMP!**
cPanel Conference '10

Pluggable Authentication

OCTOBER 4TH-6TH, 2010

```
graph LR; Service[Service] --> cPanelCode[cPanel Code]
```

cPanel

It's also possible for a services to communicate directly with an authentication binary either over a socket or by running a program.



cPanel auth binary:

- Pure-FTPD

The only example of the setup at present is pure-ftpd.

**AUTOMATION
BOOTCAMP!**
cPanel Conference '10

Pluggable Authentication

OCTOBER 4TH-6TH, 2010

```
graph LR; Service[Service] --> AuthServer[Auth Server]; AuthServer --> cPanelCode[cPanel Code];
```

cPanel

And lastly we have some services that have a server to manage authentication attempts. These are hooked into cPanel code by running a cPanel provided binary.



Auth Server:

- Dovecot
- Courier
- EXIM

Examples of this setup are dovecot and courier. EXIM is also integrated this way by using either the dovecot or courier authentication servers.



Now lets look at some of the major quirks with the way the current authentication system works.



Quirks: Virtual Accounts

Virtual accounts are very quirky.



Quirks: Virtual Accounts

- Not a first class concept

The main reason is that virtual accounts are not a first class concept on cPanel systems. Each service that handles virtual accounts does so independently from other services.

AUTOMATION

BOOTCAMP!

cPanel Conference '10

Pluggable Authentication

OCTOBER 4TH-6TH, 2010

Quirks: Virtual Accounts

- Not a first class concept
- No unified management

cPanel

There is no virtual account management interface in cPanel.

Create virtual FTP user does not create virtual webdav user

Likewise with virtual mail accounts.



Quirks: Virtual Accounts

- Not a first class concept
- No unified management
- Different ways of specifying virtual accounts

The decision of whether or not a login is virtual is also quirky.

Generally [user@domain.com](#) is virtual, but this is not always the case.



Quirks: Virtual Accounts

- Not a first class concept
- No unified management
- Different ways of specifying virtual accounts
- Confusing backend implementation

The confusion about virtual accounts is not just an interface issue.

- Hashed passwords stored in many different locations
- Code that implements virtual logins is not unified

AUTOMATION
BOOTCAMP!
cPanel Conference '10

Pluggable Authentication

OCTOBER 4TH-6TH, 2010

Quirks: Override Logins

cPanel

The next really quirky area I'd like to point out is the override login system.



Quirks: Override Logins

Example: Reseller logs into cPanel user account with reseller password

Example: cPanel user logs into virtual mail account with cPanel user password

Override logins allow you to log in to an account using the password of a more privileged user:

Example: reseller → cpanel user

Example: cpanel user → virtual mail

AUTOMATION
BOOTCAMP!
cPanel Conference '10

Pluggable Authentication

OCTOBER 4TH-6TH, 2010

Quirks: Override Logins

- Confusing rules

cPanel

The rules that apply to override logins are very confusing.

Ex: Root password can log in to a cpanel user in the cpanel interface but not the cpanel user in the webmail interface.

Ex: Root password can't be used in an override login to webmail

Ex: cPanel password can't be used in an override login to mailman



Quirks: Override Logins

- Confusing rules
- Some services unsupported

Many services where you'd expect to see override logins don't actually support them.



Quirks: Override Logins

- Confusing rules
- Some services unsupported
- Resellers with “all” privs

Resellers with “all” privileges are treated just like root in some circumstances and in others they are given no special treatment.



Quirks: Override Logins

- Confusing rules
- Some services unsupported
- Resellers with “all” privs
- Settings scattered

Like virtual accounts, the toggles that control override logins are scattered around WHM.

AUTOMATION

BOOTCAMP!

cPanel Conference '10

Pluggable Authentication

OCTOBER 4TH-6TH, 2010

Quirks: Access Controls

cPanel

Access controls are also somewhat quirky under the current implementation.



Quirks: Access Controls

- cPHulk

CPHulk is the most ubiquitous access control mechanism on cPanel systems, but it's still not perfect.



Quirks: Access Controls

- cPHulk
- Courier

With courier, for instance, cPHulk can't get IP addresses for incoming connections and can only ban on the basis on attempted usernames.



Quirks: Access Controls

- cPHulk
- Courier
- ProFTPD

ProFTPD had no cPHulk integration at all.

If you want hulk protection for FTP you have to use Pure-FTP at present.



Quirks: Access Controls

- cPHulk
- Courier
- ProFTPD
- Mailman/MySQL/PostgreSQL

Mailman, MySQL and PostgreSQL are only protected when they are accessed through cPanel.

To get complete Hulk protection for these services you have to disable any other interfaces into them.

AUTOMATION

BOOTCAMP!

cPanel Conference '10

Pluggable Authentication

OCTOBER 4TH-6TH, 2010

Quirks: Access Controls

- cPHulk
- Courier
- ProFTPD
- Mailman/MySQL/PostgreSQL
- Customization?

cPanel

Hulk is good, but what if you want to write your own code that sits at the junction Hulk presently does?

You could do something like fail2ban that parses logs and blocks at the firewall, but this is a somewhat brittle solution.

AUTOMATION

BOOTCAMP!

cPanel Conference '10

Pluggable Authentication

OCTOBER 4TH-6TH, 2010

Quirks: Logging

cPanel

Authentication logging is a very quirky area itself.



Quirks: Logging

- Auth logging currently scattered

The logs of authentication attempts are scattered around in different service log files.



Quirks: Logging

- Auth logging currently scattered
- Success vs Failure

Some services log only success, some log only failure.

EX: cpsrvd's login_log



Quirks: Logging

- Auth logging currently scattered
- Success vs Failure
- Username vs Remote IP

Some services log the attempted username for failed logins, some log only the remote IP.

AUTOMATION

BOOTCAMP!

cPanel Conference '10

Pluggable Authentication

OCTOBER 4TH-6TH, 2010

Quirks: Password Hashes

cPanel

The use of hashed passwords in the present system is another quirky area.



Quirks: Password Hashes

- System expects salted password hashes

At present all forms of authentication on cPanel systems are dependent on the validation of salted password hashes.



Quirks: Password Hashes

- System expects salted password hashes
- HTTP Digest

We don't keep the plaintext password, which makes supporting HTTP Digest authentication very difficult.



Quirks: Password Hashes

- System expects salted password hashes
- HTTP Digest
- Minimal trust of sessions

Although we do use sessions, the sessions are not given a central role in the auth system and we end up revalidating the passwords in many instances where sessions could be used.



Quirks: Password Hashes

- System expects salted password hashes
- HTTP Digest
- Minimal trust of sessions
- Password hashes must be on server

The hashed passwords must be kept on the system so that they can be validated. This is a large impediment in implementing a centralized authentication system like LDAP or RADIUS.



The design of the current auth system also has a large impact on Quality control and development.



Quirks: QA/Development

- Auth implementations are independent

Each service tends to carry around its own set of authentication code that is similar but not entirely identical to the other services on the system.

AUTOMATION

BOOTCAMP!

cPanel Conference '10

Pluggable Authentication

OCTOBER 4TH-6TH, 2010

Quirks: QA/Development

- Auth implementations are independent
- Difficult to test

cPanel

This makes it very difficult for QA to verify every authentication codepath in every authenticating service.

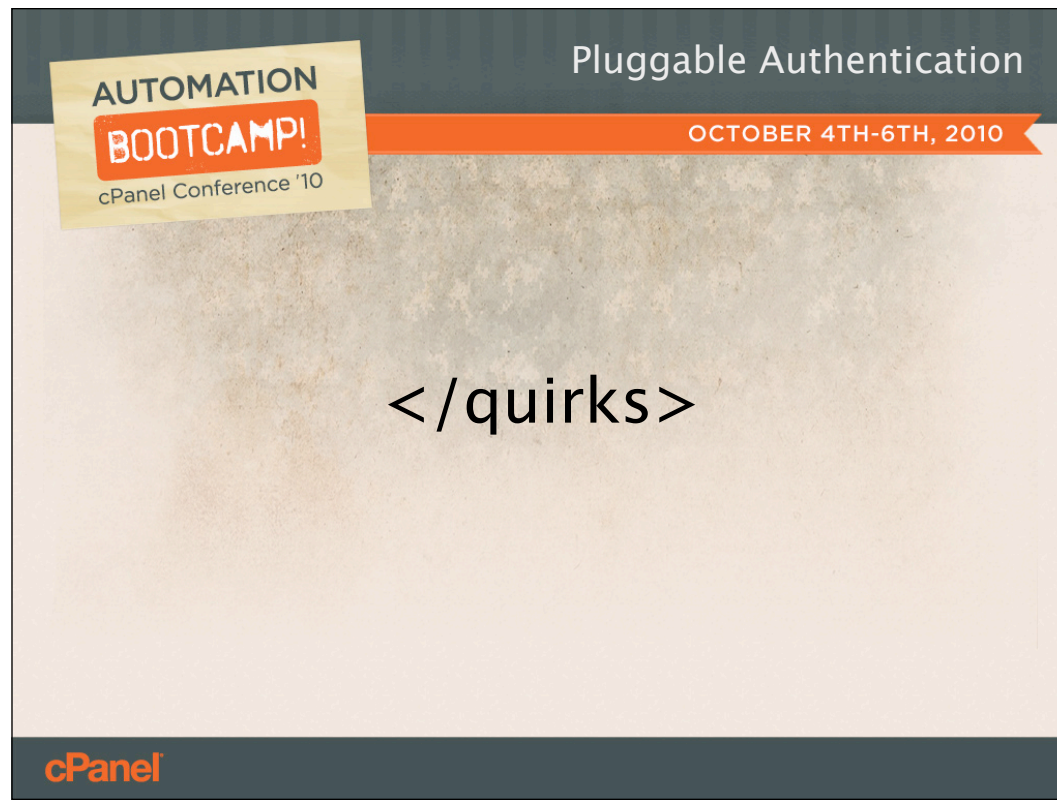


Quirks: QA/Development

- Auth implementations are independent
- Difficult to test
- Difficult for customers to replicate

It also makes it nearly impossible for a customer to validate credentials in an identical fashion to other cPanel services.

You need to write your own auth implementation from scratch.

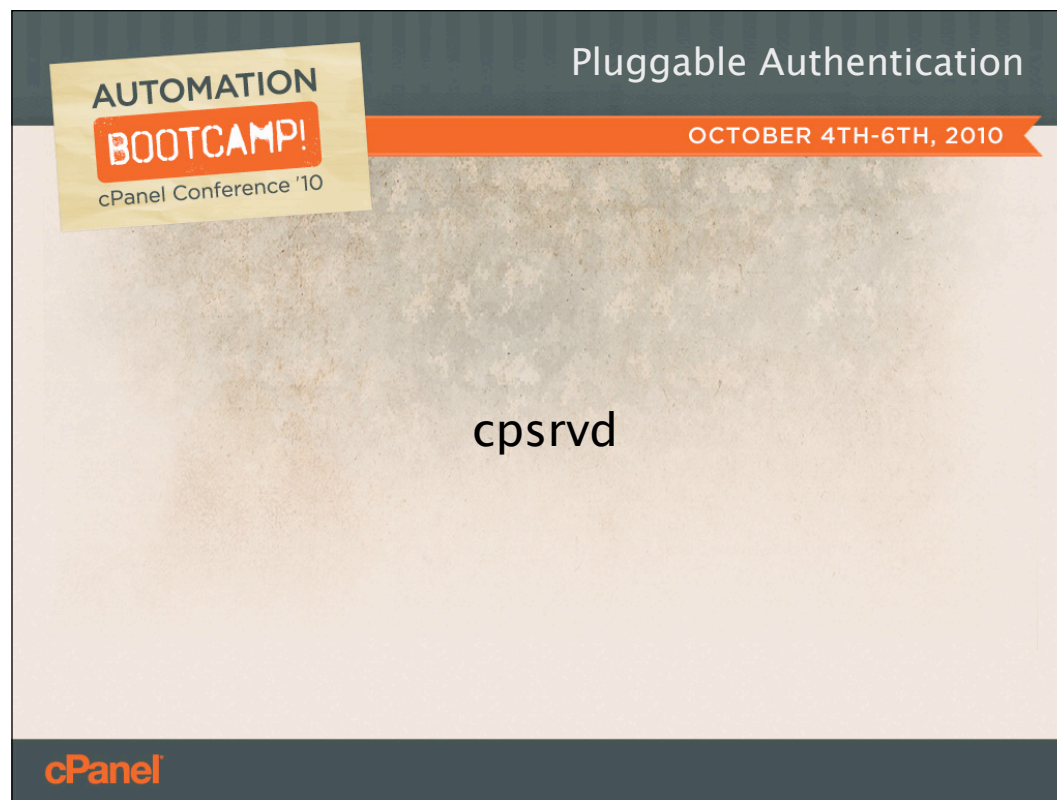


That's a lot to take in.

There are definitely rough edges on the existing authentication implementation.



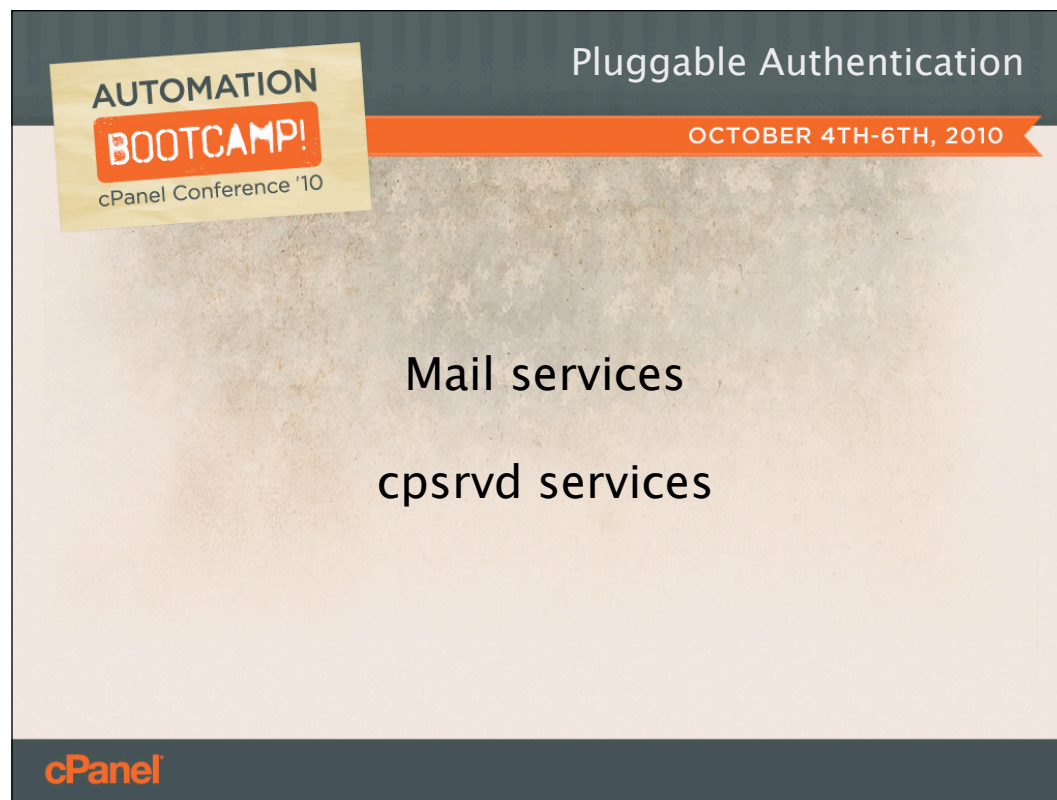
There are also some great features of the existing implementation though.



Cpsrzd is the most important.

Providing cPanel and WHM type services through Apache and Suexec would be a nightmare to secure.

cPSrzd is also central to the existing override login system.



The existing implementation also has two clusters of services that are fairly consistent and have great integration.

These provide us with an excellent starting point.



We also have nice protections for authentication that have been implemented on top of cpsrvd.



Auth Protections

- Security tokens

Security tokens were recently added and they provide excellent protection against CSRF and cookie theft attacks.



Auth Protections

- Security tokens
- Disable HTTP auth

The option to disable HTTP authentication has been in the product for some time now and it's definitely a good idea to do so.



Auth Protections

- Security tokens
- Disable HTTP auth
- cPHulk

I mentioned cPHulk already. Outside of its quirks, it's an excellent form of brute force protection to enable.



Auth Protections

- Security tokens
- Disable HTTP auth
- cPHulk
- Cookie checks

We also have the precursors of the security tokens still available in the Cookie checks



Auth Protections

- Security tokens
- Disable HTTP auth
- cPHulk
- Cookie checks
- Referrer Checks

And the referrer checks



Auth Protections

- Security tokens
- Disable HTTP auth
- cPHulk
- Cookie checks
- Referrer Checks
- Require SSL

You can force your users to use SSL when accessing cPanel and WHM now.



Auth Protections

- Security tokens
- Disable HTTP auth
- cPHulk
- Cookie checks
- Referrer Checks
- Require SSL
- SSL crypto controls

And you can thoroughly control the crypto schemes that are used with SSL for most services.



Lastly, for services that are well integrated with cPanel's existing authentication system, chkservd will perform test logins to verify the service is really functional.

This is a great feature we want to see added across the board.



Pluggable Authentication

OCTOBER 4TH-6TH, 2010

AUTOMATION

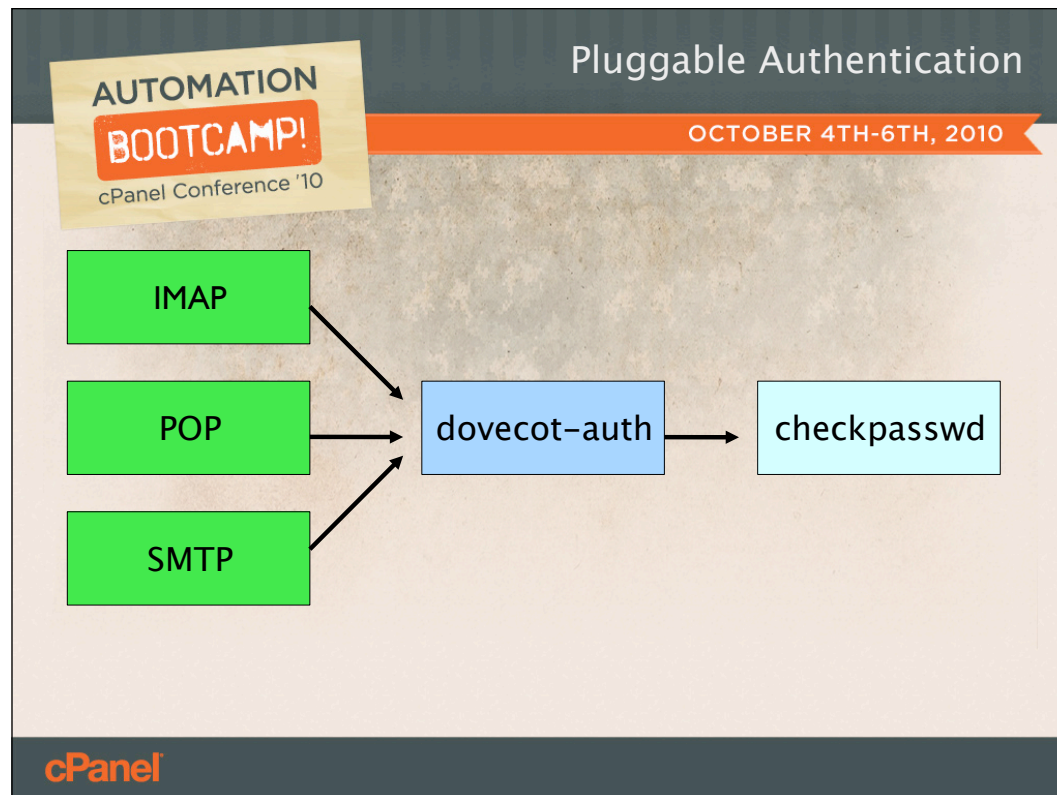
BOOTCAMP!

cPanel Conference '10

Roadmap to Pluggable Auth

cPanel

So lets shift our focus now from the existing authentication system to the overall design of the new system.

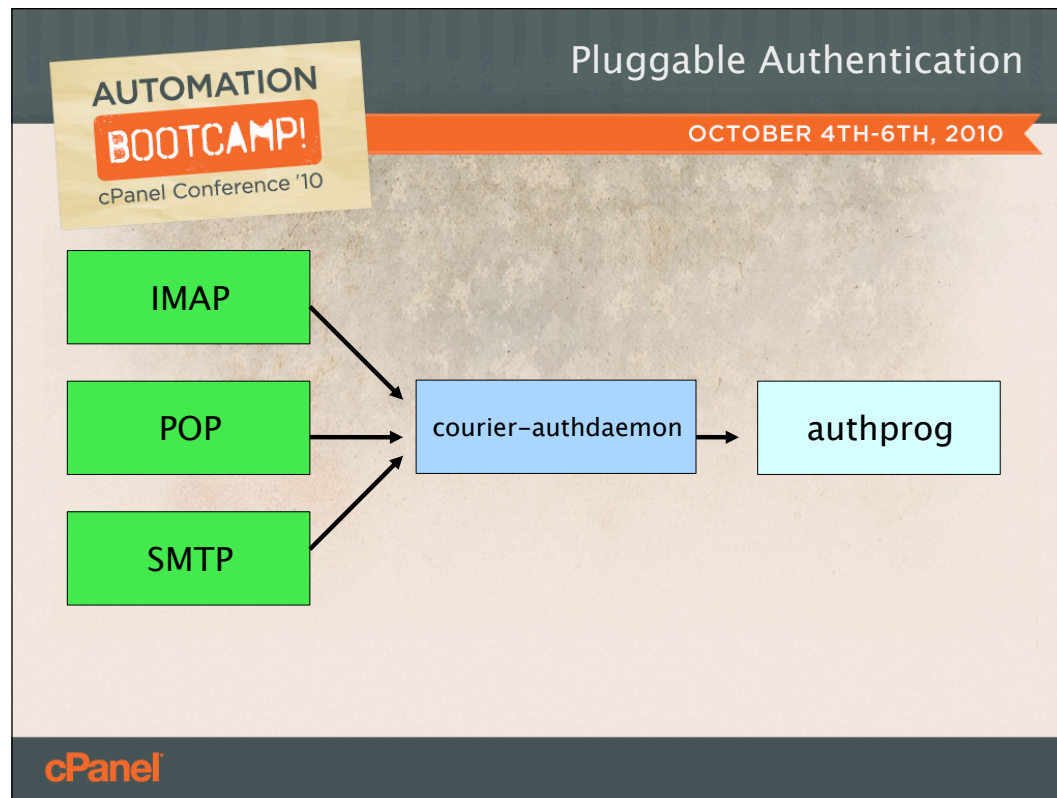


Dovecot

Imap talks to dovecot-auth server

That talks to a cpanel checkpassword binary

That runs cpanel auth code

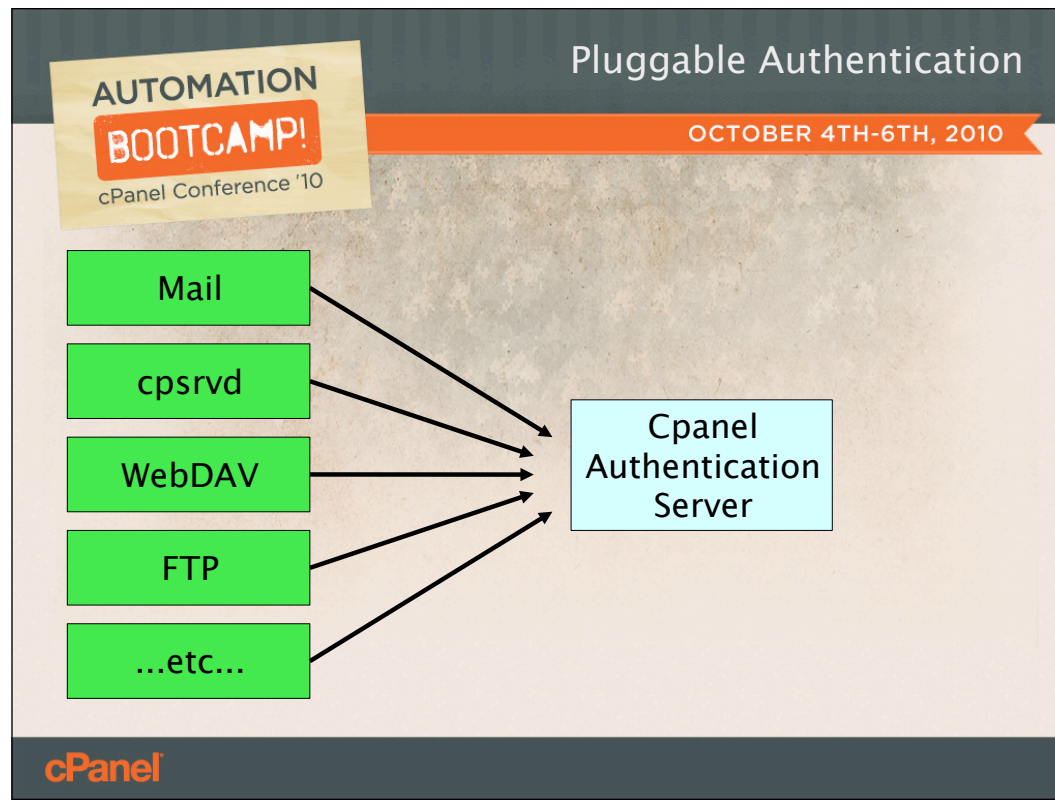


Courier is nearly identical

Imap talks to courier-authdaemon server

That talks to a cpanel authprog binary

That runs cpanel auth code



The new system will be very similar. We're going to replace courier-authdaemon and the dovecot-auth server with our own auth server system.

We're also going to hook all of the existing services that authenticate users into the new cpanel auth server system.

AUTOMATION
BOOTCAMP!
cPanel Conference '10

Pluggable Authentication

OCTOBER 4TH-6TH, 2010

Cpanel
Authentication
Server

- Caching

cPanel

The new system will allow us to cache files that currently have to be reread each time an authentication attempt is made.

AUTOMATION
BOOTCAMP!
cPanel Conference '10

Pluggable Authentication

OCTOBER 4TH-6TH, 2010

Cpanel
Authentication
Server

- Caching
- Memory impact

cPanel

Since we're removing either dovecot-auth or courier-authdaemon, the memory impact on most systems should be negligible.

Hopefully any additional memory used by the cpanel auth server will be made up in the servers that use it having less bloat to carry around.

**AUTOMATION
BOOTCAMP!**
cPanel Conference '10

Pluggable Authentication

OCTOBER 4TH-6TH, 2010

Cpanel
Authentication
Server

- Caching
- Memory impact
- Performance impact

cPanel

The caching available through having a persistent authentication server should improve the performance of most systems that need to do authentication.

Most setup will already be done before an individual authentication attempt takes place. Services will just connect with a socket and authenticate.

AUTOMATION
BOOTCAMP!
cPanel Conference '10

Pluggable Authentication

OCTOBER 4TH-6TH, 2010

Cpanel Auth Server != dovecot-auth

cPanel

If you're familiar with dovecot-auth you might ask why we don't just standardize on it and hook all of the other services into dovecot-auth.

There are a few good reasons for replacing it.



Cpanel Auth Server != dovecot-auth

- Service specific data

Different services need different types of authentication data.

Ex: FTP quota vs Mail quota



Cpanel Auth Server != dovecot-auth

- Service specific data
- Multi-protocol

We also want a server that can handle multiple communication protocols. If we standardized on dovecot's protocol we would have a very difficult time getting courier-imap to authenticate against it.

AUTOMATION

BOOTCAMP!

cPanel Conference '10

Pluggable Authentication

OCTOBER 4TH-6TH, 2010

Cpanel Auth Server == autofixer

cPanel

This change to a central auth server fixes many difficult problems by itself.

AUTOMATION

BOOTCAMP!

cPanel Conference '10

Pluggable Authentication

OCTOBER 4TH-6TH, 2010

Cpanel Auth Server == autofixer

- Override synchronization

cPanel

The synchronization of override logins becomes a given.

We actually have to worry about where we want the existing quirks to stay in place.



Cpanel Auth Server == autofixer

- Override synchronization
- cPHulk

CPHulk support also becomes a given. It's handled by the auth server and doesn't need to be implemented by the authenticating service.



Cpanel Auth Server == autofixer

- Override synchronization
- cPHulk
- Sessions

Sessions become centralized and we can use them across services in ways that are difficult to implement now.

Ex: Passwords are currently passed through to the webmail applications so that they can send them on to IMAP and SMTP. With the new design we can set up a session at the auth server level and have it accepted by all services.

AUTOMATION

BOOTCAMP!

cPanel Conference '10

Pluggable Authentication

OCTOBER 4TH-6TH, 2010

Cpanel Auth Server == maintainability

cPanel

The centralized authentication server will greatly improve the maintainability of the authentication system.



Cpanel Auth Server == maintainability

- Object Oriented

The new system is Object oriented from the get-go.



Cpanel Auth Server == maintainability

- Object Oriented
- Testable

Individual modules are testable and the system as a whole is testable.

AUTOMATION

BOOTCAMP!

cPanel Conference '10

Pluggable Authentication

OCTOBER 4TH-6TH, 2010

Cpanel Auth Server == maintainability

- Object Oriented
- Testable
- Centralized

cPanel

All of the logic is centralized in one place instead of being scattered in each service.

EX: 15% of the code in cpsrvd is general authentication logic.

The quantity of code required to implement many features of the present system is one reason why so many services do not implement all of the features.



Cpanel Auth Server == maintainability

- Object Oriented
- Testable
- Centralized
- Isolated

And we're also isolating the authentication code from the service code.

We can fix bugs in the authentication server without worrying about unintended side effects in unrelated service functionality.



This is called the PLUGGABLE authentication framework though, right?

I haven't mentioned extensibility much yet.



In reality, the benefits of the new system that I've discussed so far were not the driving motivation behind this project.

AUTOMATION

BOOTCAMP!

cPanel Conference '10

Pluggable Authentication

OCTOBER 4TH-6TH, 2010

Wishlist

- Hook into existing auth systems

cPanel

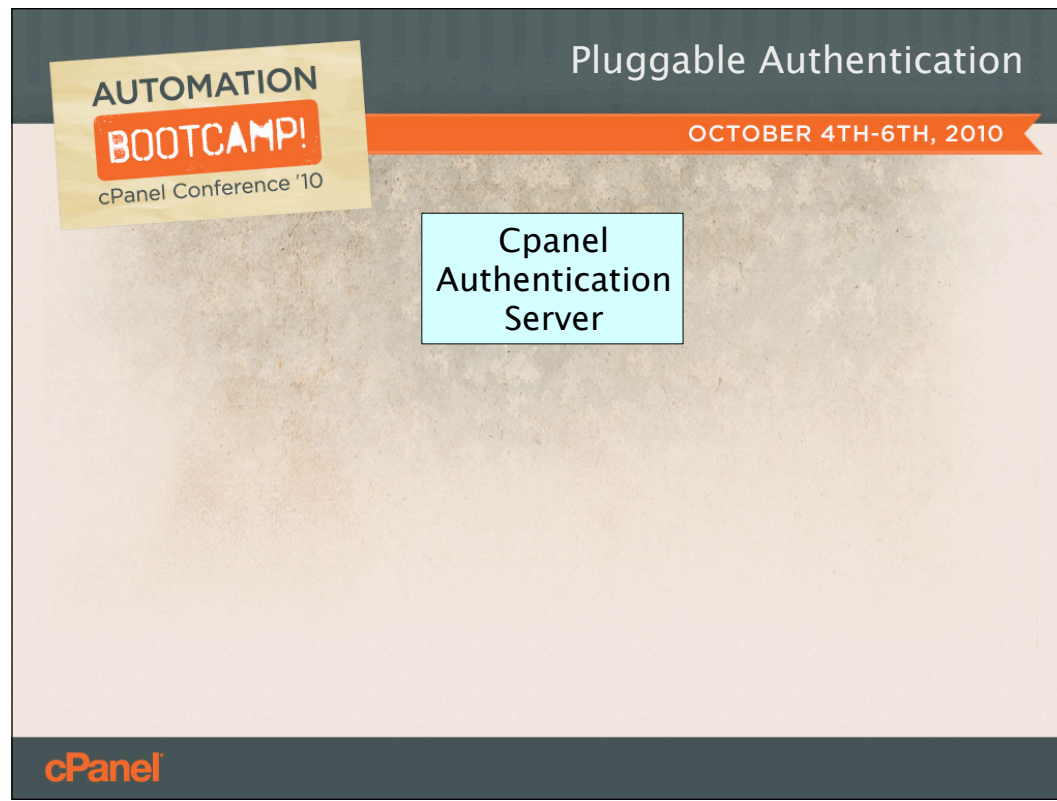
Many customers wanted to hook cpanel's auth system into their existing centralized authentication servers like LDAP.



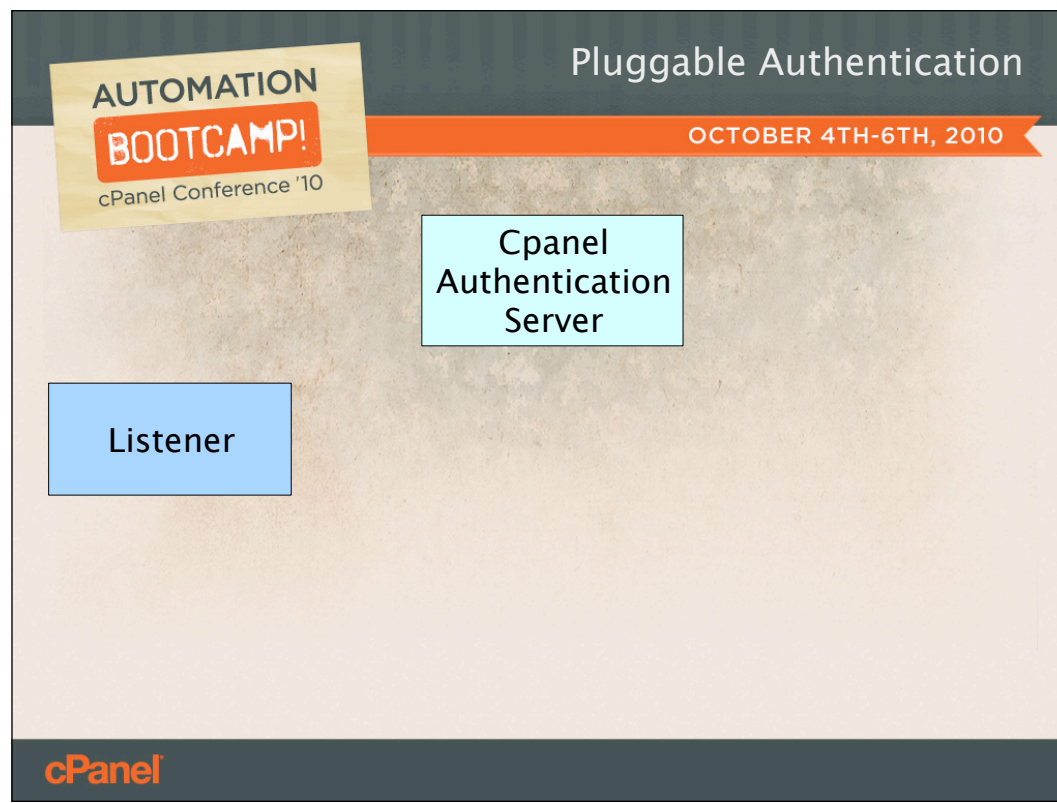
Wishlist

- Hook into existing auth systems
- Custom cPHulk systems

Some customers also wanted the ability to hook simple checks into every place that authentication happens in the same way that cPHulk does.



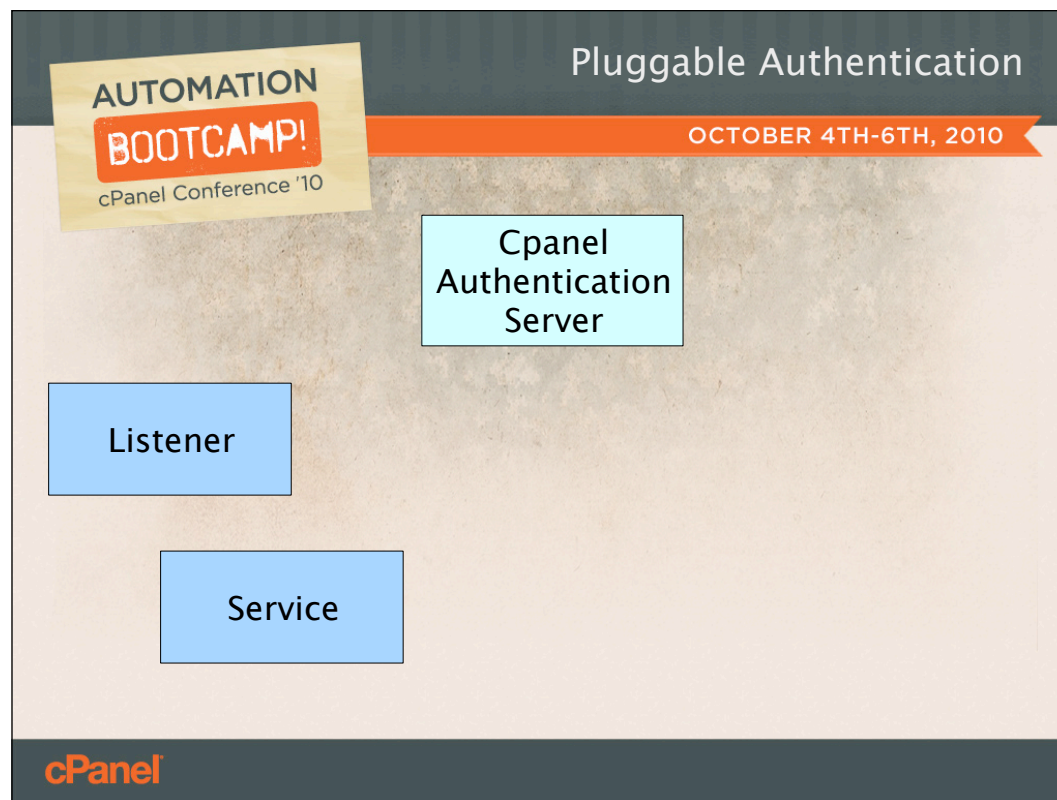
The modular design of the authentication server addresses these needs.



To start with we have Listener modules that communicate different protocols on different sockets.

Ex: courier listener vs dovecot listener

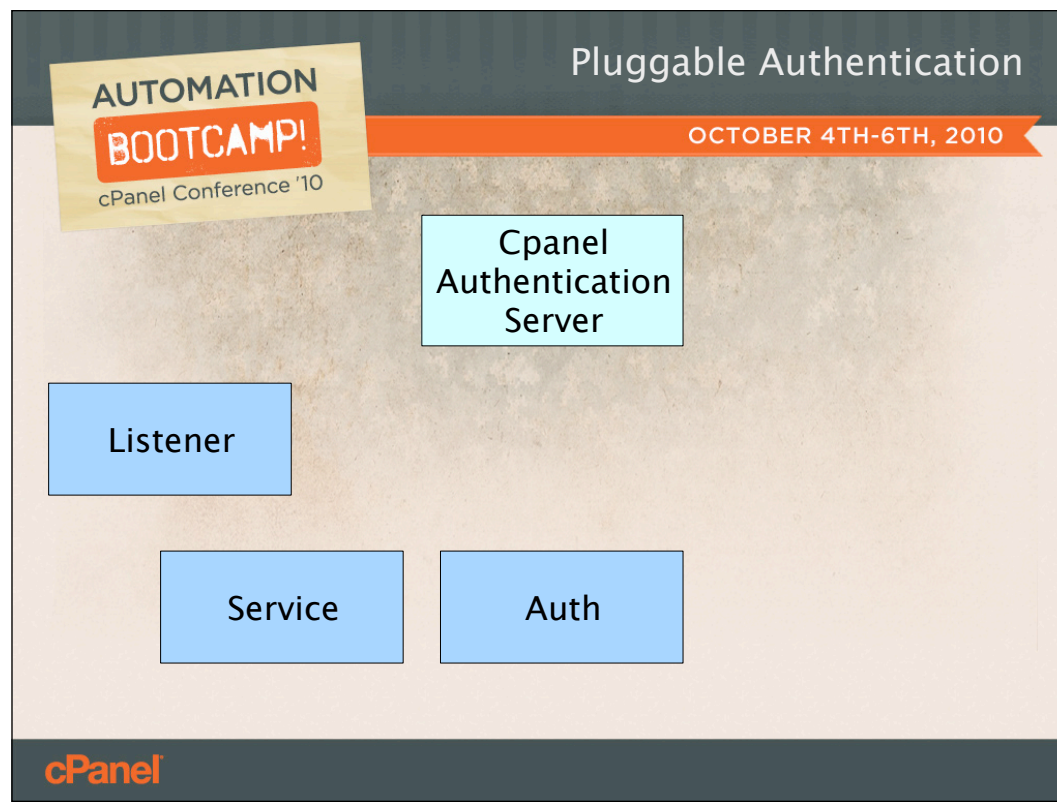
Use: If you have existing services that authenticcate over a socket you may save effort by writing a custom listener. If not, you'll want to use the listeners we provide.



Service modules will encapsulate the logic specific to each service.

Ex: FTP quotas vs Mail quotas.

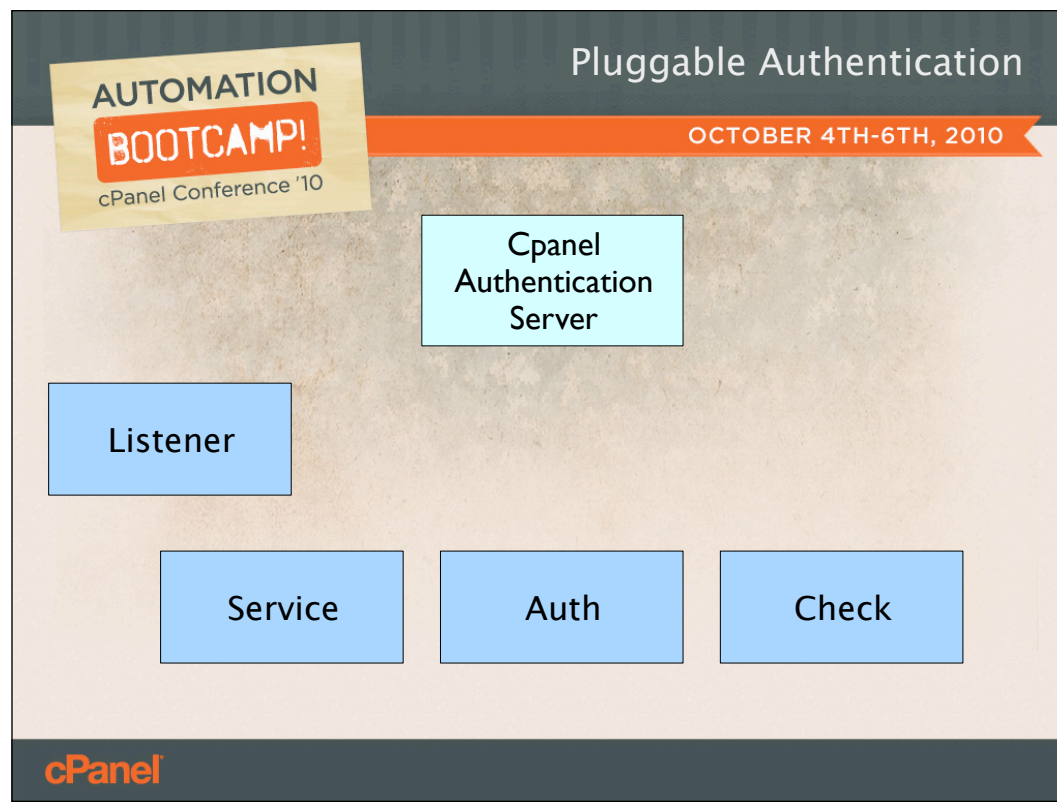
Use: Changing existing cPanel behavior like overrides in some subtle way,



Auth modules will handle different methods of authentication.

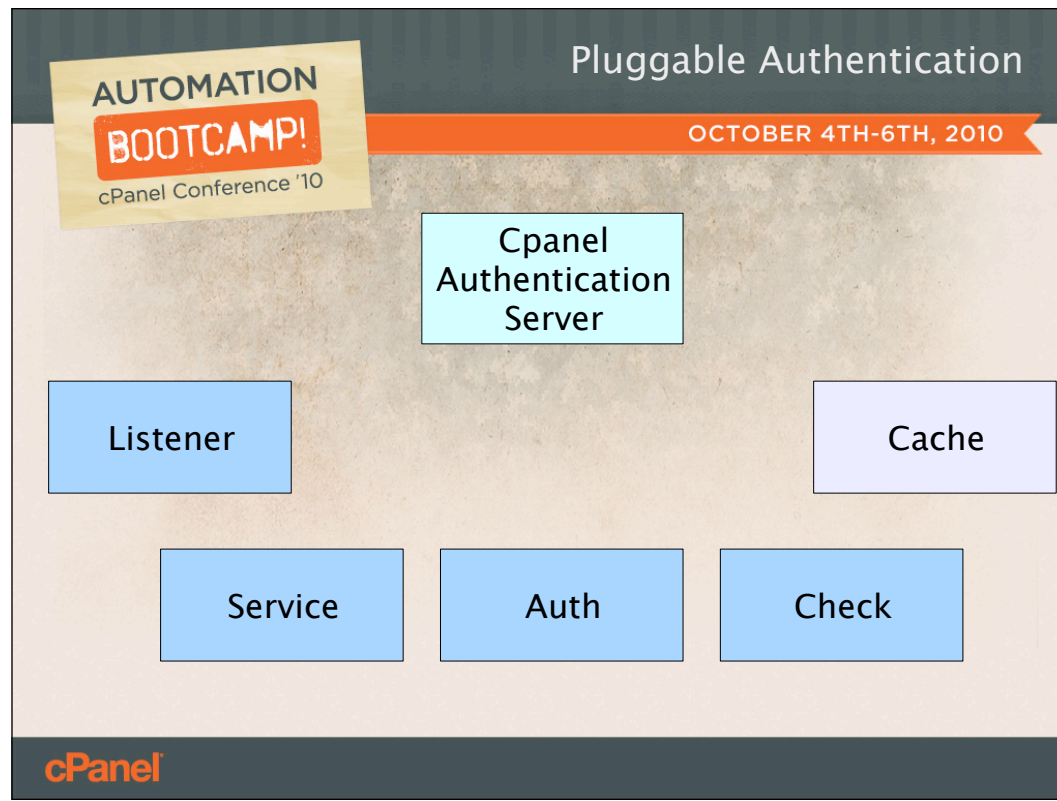
EX: virtual vs system vs cpanel

Use: To use something other than hashed passwords for authentication, this will be the place to hook in.

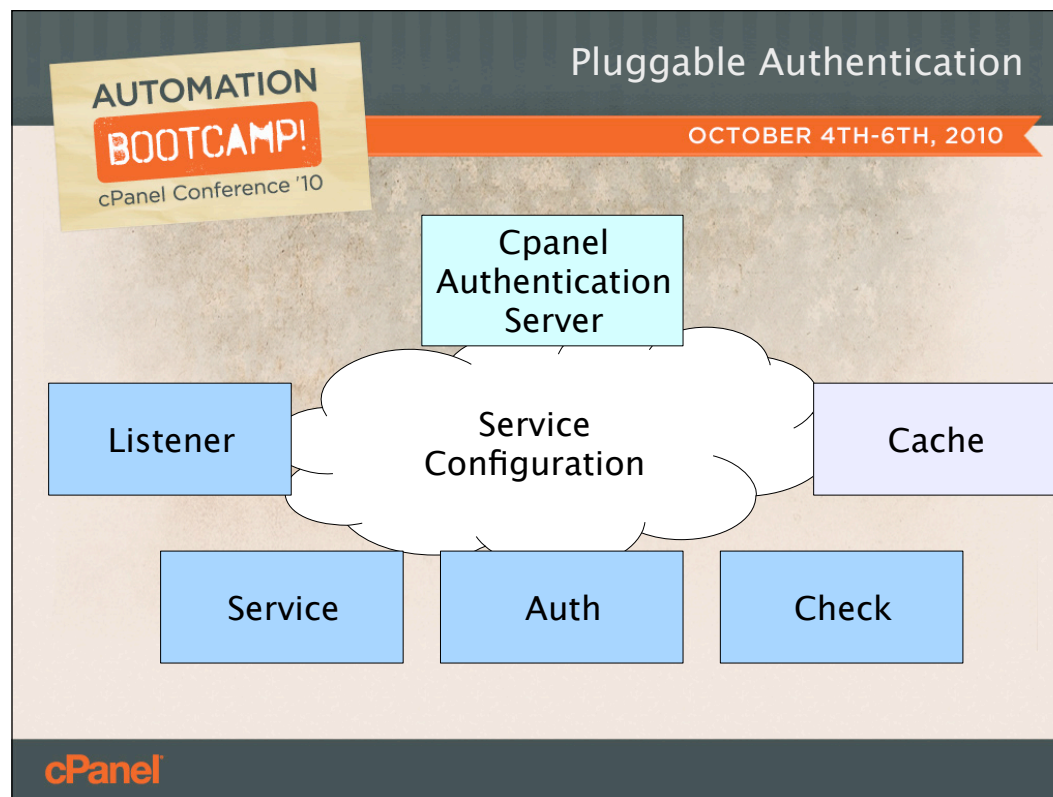


Check modules will implement straightforward Hulk and PAM style access controls that can abort an authentication before or after it has been validated.

This will be the simplest module type to hook into



Cache modules are going to manage a memory cache of system credential information. These may be limited to cPanel provided modules.



Gluing it all together will be service configuration files that specify which modules apply to each service.

This will provide the ability to do really simple customizations without writing a line of code.

Ex: Disable Hulk for FTP only

Ex: Disable Virtual account logins to SMTP

Ex: Disable all standard logins to cPanel and only use your custom modules.



That's the heart of the pluggable auth system, but there are a few other items that have been thrown in with this project that we hope to address.



I already mentioned HTTP Digest authentication.

We've tried to support it in the past without success.



HTTP Digest Authentication

- Still needed

Support is still very much needed for:

- WebDAV clients
- Certain PCI compliance scenarios

AUTOMATION
BOOTCAMP!
cPanel Conference '10

Pluggable Authentication

OCTOBER 4TH-6TH, 2010

HTTP Digest Authentication

- Still needed
- Still difficult

cPanel

HTTP Digest is still a very difficult problem to solve.

There is no way to validated Digest passwords using the salted password hashes we currently have available.

The only way to compute the HTTP digest hashes is to have access to the plaintext passwords.

AUTOMATION
BOOTCAMP!
cPanel Conference '10

Pluggable Authentication

OCTOBER 4TH-6TH, 2010

HTTP Digest Authentication

- Still needed
- Still difficult
- Slightly easier to create

cPanel

With the centralized authentication framework it's SLIGHTLY easier to try solving this dilemma.

We can create the digest hashes inside the auth server while we have the plaintext passwords available.

The centralization of the new system makes it less work to implement.

We can also validate the authentication attempt before the auth server performs the digest hashing in a separate process.



Cleaning up the virtual account systems is something I'm keenly interested in.



Virtual Account Cleanup

- Single account

What we really want is a single virtual account....



Virtual Account Cleanup

- Single account
- Multiple services

Having access to a configurable list of servers.



Virtual Account Cleanup

- Single account
- Multiple services
- Synchronization?

The difficult obstacle to overcome is synchronizing all of the existing disconnected virtual accounts into unified virtual account entities.

How can we be certain that the foo@example.com mail account is really the same person as the foot@example.com webdav account?



Virtual Account Cleanup

- Single account
- Multiple services
- Synchronization?
- Long range benefits

If we can work this out there are some excellent long range possibilities this opens up such as restricted cpanel logins for virtual accounts.

**AUTOMATION
BOOTCAMP!**
cPanel Conference '10

Pluggable Authentication

OCTOBER 4TH-6TH, 2010

Improved Service Integration

- SFTP virtual accounts
- IP logging/banning
- Full auth integration

cPanel

Improved service integration is also a project goal.

SFTP virtual logins would be awesome

IP logging and banning for all services is a major goal for the project.

Full integration with some of the more independent services like mailman and mysql would also be wonderful.

AUTOMATION

BOOTCAMP!

cPanel Conference '10

Pluggable Authentication

OCTOBER 4TH-6TH, 2010



cPanel

Now we're at the end of the talk.

As I said at the beginning, this is a work in progress.

AUTOMATION
BOOTCAMP!
cPanel Conference '10

Pluggable Authentication

OCTOBER 4TH-6TH, 2010

Ask yourself:

cPanel

You can have a huge impact on the system. Just ask yourself

AUTOMATION
BOOTCAMP!
cPanel Conference '10

Pluggable Authentication

OCTOBER 4TH-6TH, 2010

What did we miss?

cPanel

What did we miss?

AUTOMATION

BOOTCAMP!

cPanel Conference '10

Pluggable Authentication

OCTOBER 4TH-6TH, 2010

What sounds too complicated?

cPanel

What sounds too complicated?



What examples and documentation would you like to see?

What documentation do you need?

AUTOMATION

BOOTCAMP!

cPanel Conference '10

Pluggable Authentication

OCTOBER 4TH-6TH, 2010

What customizations of the new system
can you see yourself making?

cPanel

Where sort of customizations can you see yourself making?



John Lightsey

jd@cpanel.net



We want to hear from you, so please feel free to email me or grab me in the halls here and bend my ear.