

# **dynalogin**

*Open source two-factor authentication*

Daniel Pocock

`daniel@pocock.com.au`

<http://www.dynalogin.org>

# Why one-time-passwords?

spam brings users to phishing sites

# Why one-time-passwords?

**spam** brings users to phishing sites

**keyloggers** in hardware and software, network sniffing, and other methods can easily compromise passwords

# Why one-time-passwords?

**spam** brings users to phishing sites

**keyloggers** in hardware and software, network sniffing, and other methods can easily compromise passwords

**password recovery** is often easily abused with data from facebook

# Why one-time-passwords?

**spam** brings users to phishing sites

**keyloggers** in hardware and software, network sniffing, and other methods can easily compromise passwords

**password recovery** is often easily abused with data from facebook

**smart cards** are not for everyone and require hardware on every PC

# Why one-time-passwords?

**spam** brings users to phishing sites

**keyloggers** in hardware and software, network sniffing, and other methods can easily compromise passwords

**password recovery** is often easily abused with data from facebook

**smart cards** are not for everyone and require hardware on every PC

**one-time-passwords** are a viable solution

# What was out there?

**RSA tokens** for those situations where money is no object

# What was out there?

**RSA tokens** for those situations where money is no object

**HOTP** an open standard for event-based one time  
passwords



# What was out there?

**RSA tokens** for those situations where money is no object

**HOTP** an open standard for event-based one time passwords

**hotp-toolkit** (now OATH toolkit) library implementing OATH standard HOTP, command line utility and PAM module based on flat files

# What was out there?

**RSA tokens** for those situations where money is no object

**HOTP** an open standard for event-based one time passwords

**hotp-toolkit** (now OATH toolkit) library implementing OATH standard HOTP, command line utility and PAM module based on flat files

**barada** similar to hotp-toolkit, PAM module and soft token

# What was out there?

**RSA tokens** for those situations where money is no object

**HOTP** an open standard for event-based one time passwords

**hotp-toolkit** (now OATH toolkit) library implementing OATH standard HOTP, command line utility and PAM module based on flat files

**barada** similar to hotp-toolkit, PAM module and soft token

**mobile OTP** a non-HOTP solution with a soft token for various phones, and a PAM module based on flat files

# What was missing?

**database** storage of key data

# What was missing?

**database** storage of key data

**privilege separation** keys stored on same machine and  
maybe even accessible to a user process

# What was missing?

**database** storage of key data

**privilege separation** keys stored on same machine and  
maybe even accessible to a user process

**modular use cases** including OpenID, RADIUS, J2EE web  
container

# What was missing?

**database** storage of key data

**privilege separation** keys stored on same machine and  
maybe even accessible to a user process

**modular use cases** including OpenID, RADIUS, J2EE web  
container

**enterprise class** provisioning and user lifecycle management

# **dynalogin aims**

**fill the gaps** identified in the existing products



# **dynalogin aims**

**fill the gaps** identified in the existing products

**stable foundation** for further algorithms (e.g. TOTP), storage options, management tools

# **dynalogin aims**

**fill the gaps** identified in the existing products

**stable foundation** for further algorithms (e.g. TOTP), storage options, management tools

**just works** when deployed out of a package, becoming a drop-in solution for any sysadmin

# Current status

**unixODBC** storage module implemented

# Current status

**unixODBC** storage module implemented  
**HOTP** algorithm from oath-toolkit

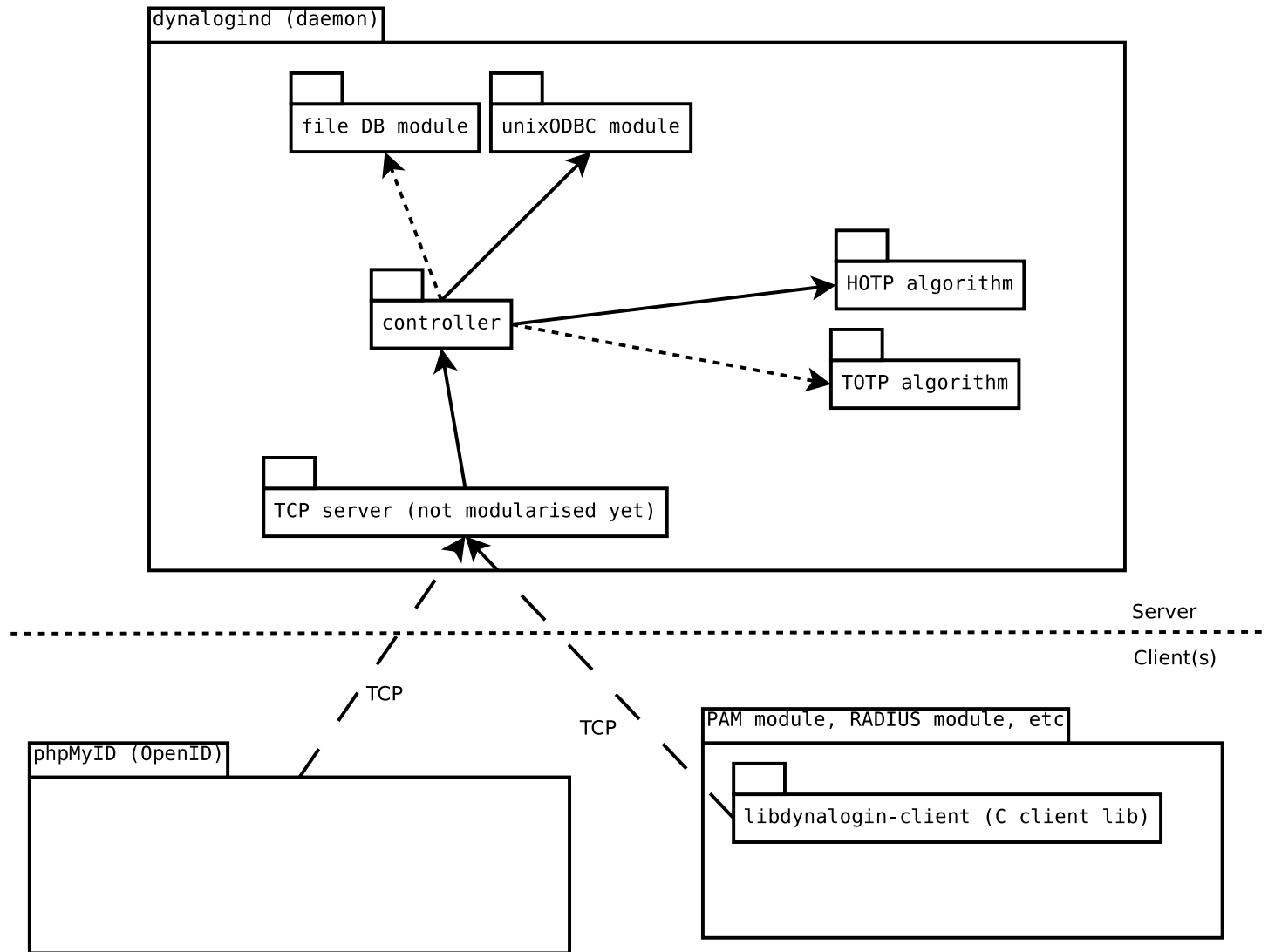
# Current status

**unixODBC** storage module implemented

**HOTP** algorithm from oath-toolkit

**OpenID** provider proof-of-concept based on phpMyID code

# Architecture diagram



# Directions

**finalise** network protocol (currently resembles SMTP command/response behavior) for the TCP client API

**generalise** for other OATH algorithms, TOTP, OCRA, ...

# Directions

**finalise** network protocol (currently resembles SMTP command/response behavior) for the TCP client API

**generalise** for other OATH algorithms, TOTP, OCRA, ...

**configurable routing** of authentication requests by the sysadmin, so that different policies can be applied based on different scenarios



# Directions

- finalise** network protocol (currently resembles SMTP command/response behavior) for the TCP client API
- generalise** for other OATH algorithms, TOTP, OCRA, ...
- configurable routing** of authentication requests by the sysadmin, so that different policies can be applied based on different scenarios
- logging** of data gathered from a higher level, e.g. name of each OpenID consumer requesting authentication

# Directions

- finalise** network protocol (currently resembles SMTP command/response behavior) for the TCP client API
- generalise** for other OATH algorithms, TOTP, OCRA, ...
- configurable routing** of authentication requests by the sysadmin, so that different policies can be applied based on different scenarios
- logging** of data gathered from a higher level, e.g. name of each OpenID consumer requesting authentication
- packaging and availability in distributions** to enable rapid and widespread deployment by sysadmins

# Routing of authentication activity

**complete modularisation** of the dynalogind server: network modules (e.g. TCP or queue based API), modular algorithms (HOTP, TOTP, OCRA), logging (file, syslog, db), key storage (file, db)

# Routing of authentication activity

**complete modularisation** of the dynalogind server: network modules (e.g. TCP or queue based API), modular algorithms (HOTP, TOTP, OCRA), logging (file, syslog, db), key storage (file, db)

**interaction** with external systems during verification, particularly for OCRA signing purposes: send data to an external system using a queue, or retrieve data from a cache

# Conclusion

**download** from `http://www.dynalopin.org` - run the  
configure script, make install

# Conclusion

**download** from `http://www.dynalogin.org` - run the  
configure script, make install

**smartphone** app available for Android, search for dynalogin  
in the market

# Conclusion

**download** from `http://www.dynalogin.org` - run the configure script, make install

**smartphone** app available for Android, search for dynalogin in the market

**try OpenID** secured by dynalogin, just copy the sample Apache config

# Conclusion

- download** from `http://www.dynalogin.org` - run the configure script, make install
- smartphone** app available for Android, search for dynalogin in the market
- try OpenID** secured by dynalogin, just copy the sample Apache config
- discuss** use cases, architecture, make contributions through the forums and mailing lists