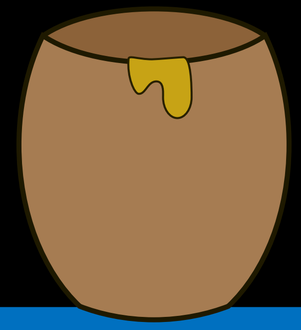
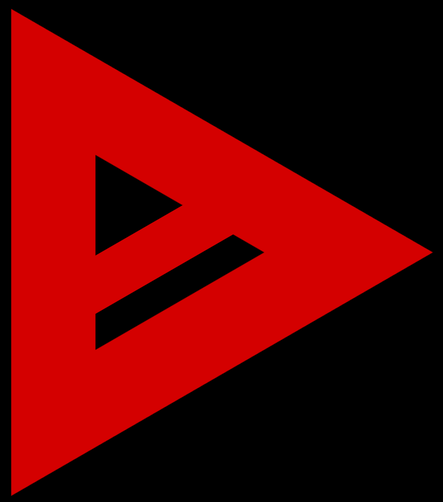
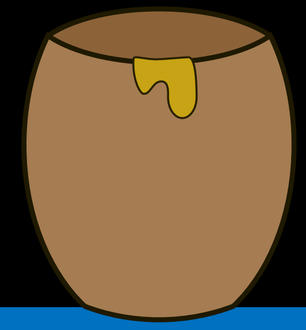
Watch the
Asciinema Replay
of Your
Home-Made Honeypot





asciinema

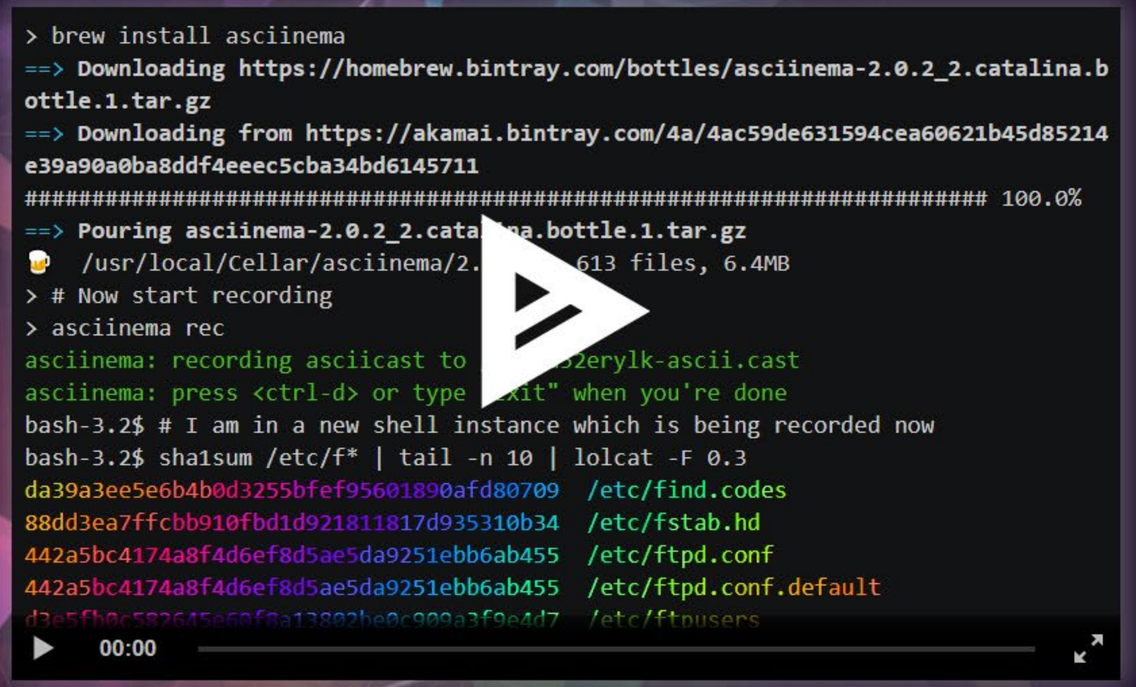


Record and share your terminal sessions, the right way.

Forget screen recording apps and blurry video. Enjoy a lightweight, purely text-based approach to terminal recording.

Start Recording

Supports Linux, macOS and *BSD



asciinema *[as-kee-nuh-muh]* is a free and open source solution for recording terminal sessions and sharing them on the web. Read about how it works.

Simple recording

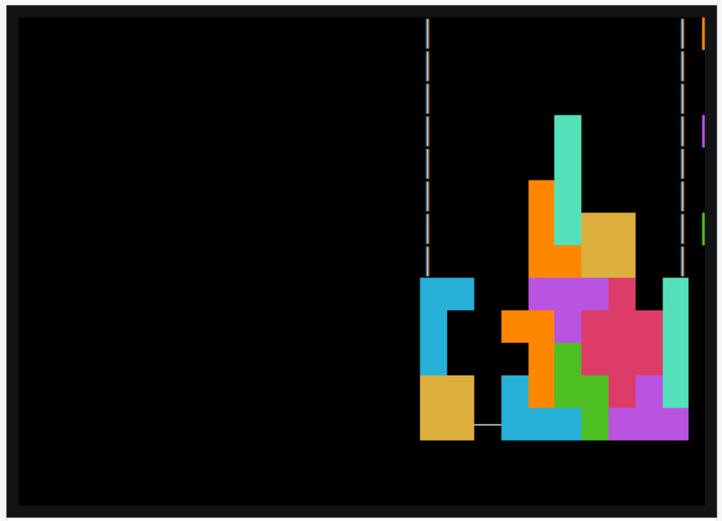
Copy & paste

Embedding

```
$ asciinema rec
asciinema: recording asciicast to /tmp/tmp2haimar6-ascii.cast
asciinema: press <ctrl-d> or type "exit" when you're done
$ echo -e "Hello FOSDEM\e[35m'21\e[0!"
Hello FOSDEM'21
$ exit
asciinema: recording finished
asciinema: press <enter> to upload to asciinema.org, <ctrl-c> to save locally

View the recording at:

https://asciinema.org/a/XSypCU7IlSWGFlVyMPP9OCoSE
```

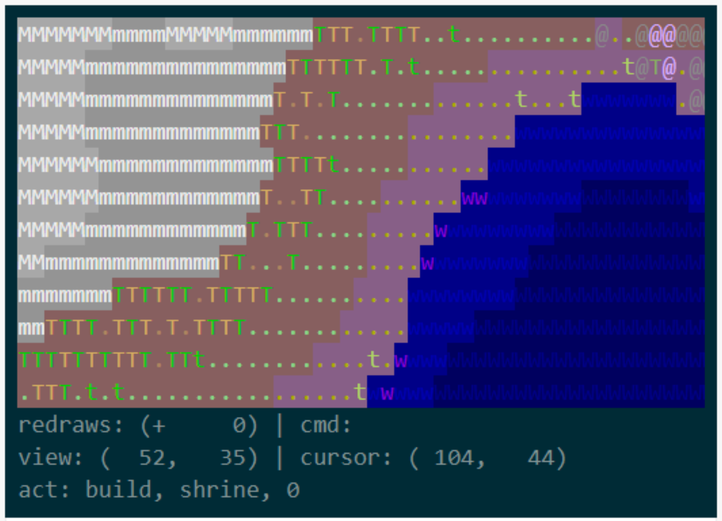
https://asciinema.org/a/386266

Example sessions recorded with asciinema



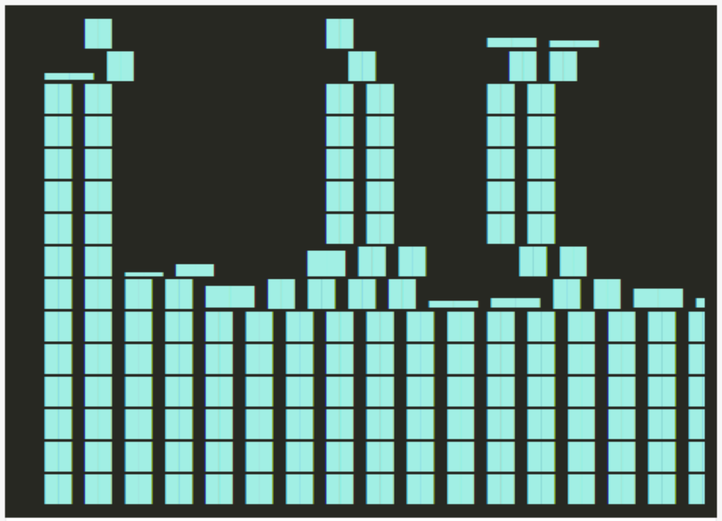
term-tris dt cannon featured 01:04

by mat44



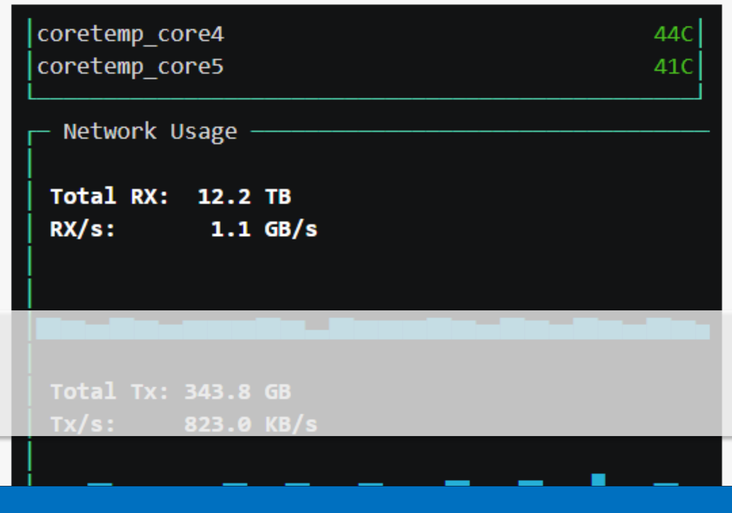
asciicast:326923 featured 04:01

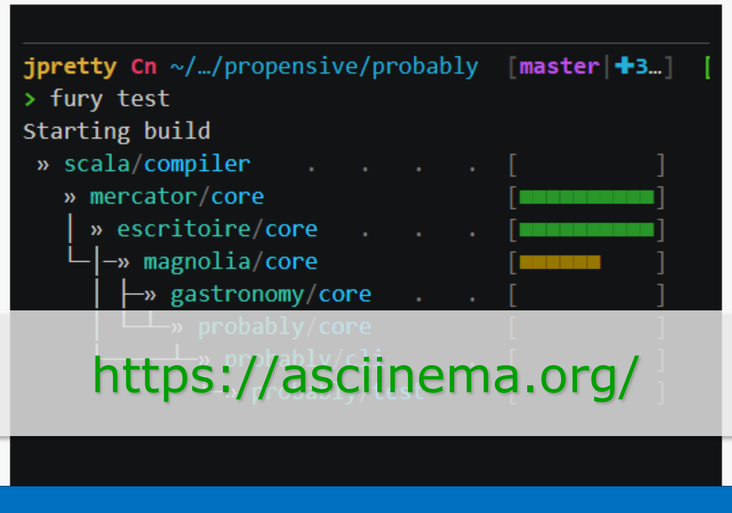
by lattis

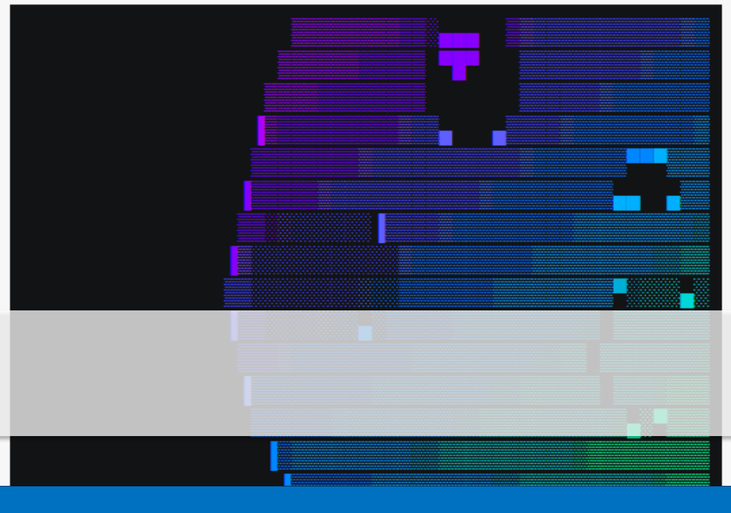


cava - alacritty featured 00:37

by fparrav







https://asciinema.org/

Photo by Art Rachen on Unsplash

asciicast v2 format

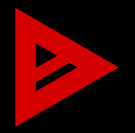
```
{
  "version": 2,
  "width": 80,
  "height": 25,
  "timestamp": 1504467315,
  "title": "Demo",
  "env": {
    "TERM": "xterm-256color",
    "SHELL": "/bin/zsh"
  }
}
```

https://blog.asciinema.org/post/two-point-o/

asciicast v2 format

```
[
  0.248848,
  "o",
  "\u001b[1;31mHello \u001b[32mWorld!\u001b[0m\n"
]
```

https://blog.asciinema.org/post/two-point-o/

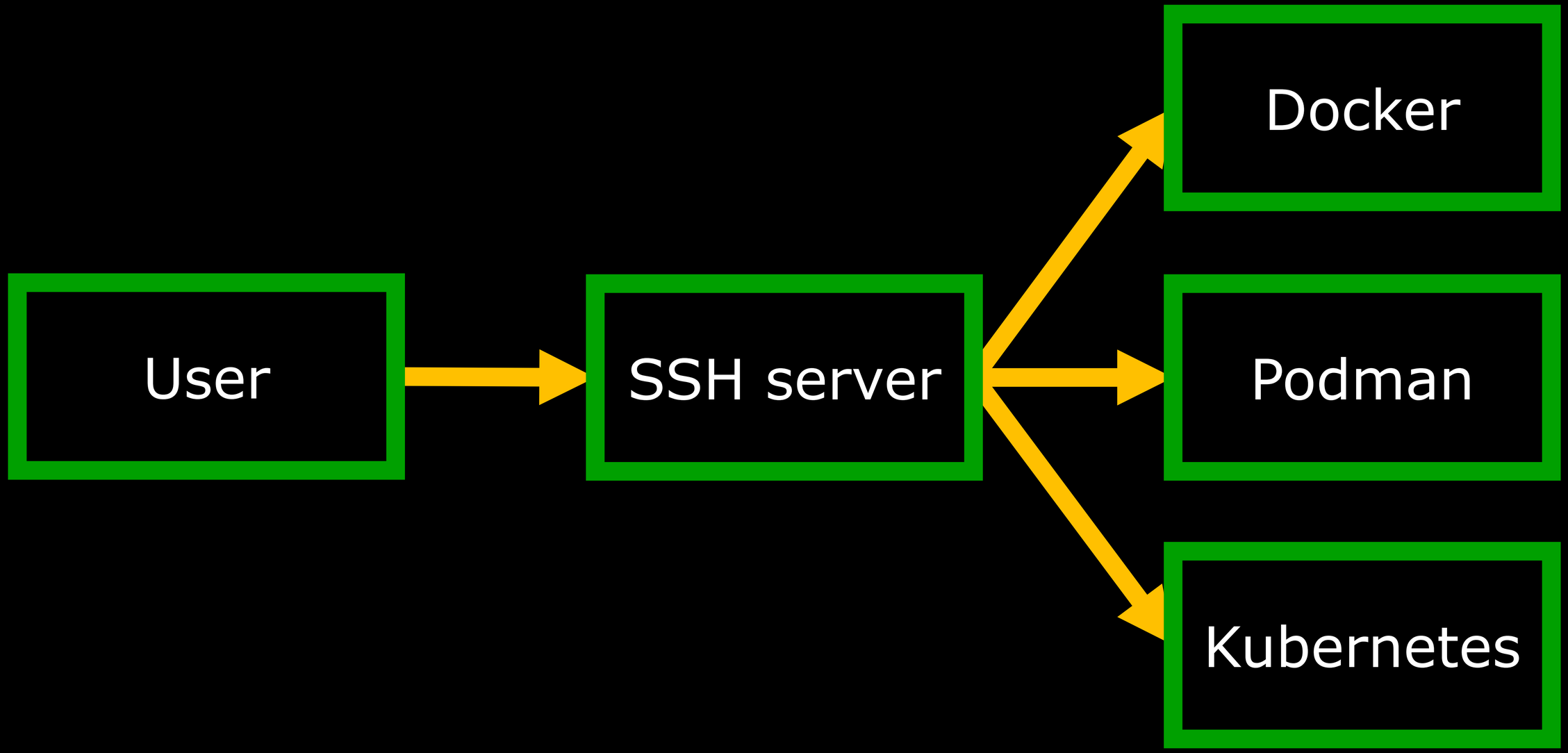


https://github.com/ContainerSSH/auditlog/

blob/main/codec/asciinema/format.go

```
# docker run -ti ubuntu
```

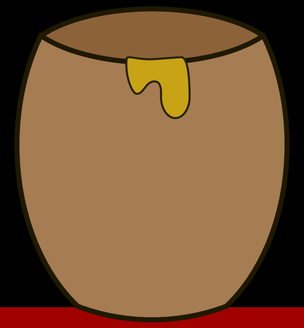
```
root@be6b26af45f1:/# echo "Hello world!"
```


User
SSH server
Docker
Podman
Kubernetes

DON’T

TRY THIS

AT HOME



/etc/ssh/sshd_config:

ForceCommand /usr/bin/docker run -ti ubuntu

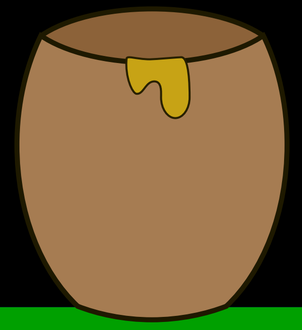
/etc/ssh/sshd_config:

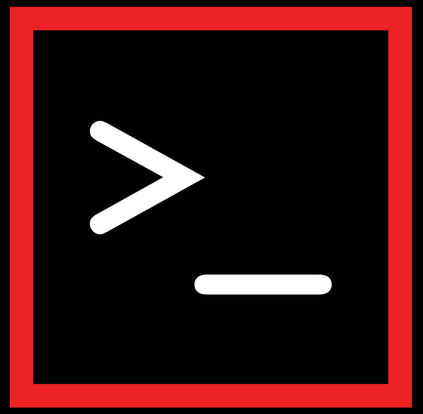
ForceCommand /usr/bin/asciinema rec /tmp/ssh.cast –c "/usr/bin/docker run -ti ubuntu"

DO

TRY THIS

AT HOME

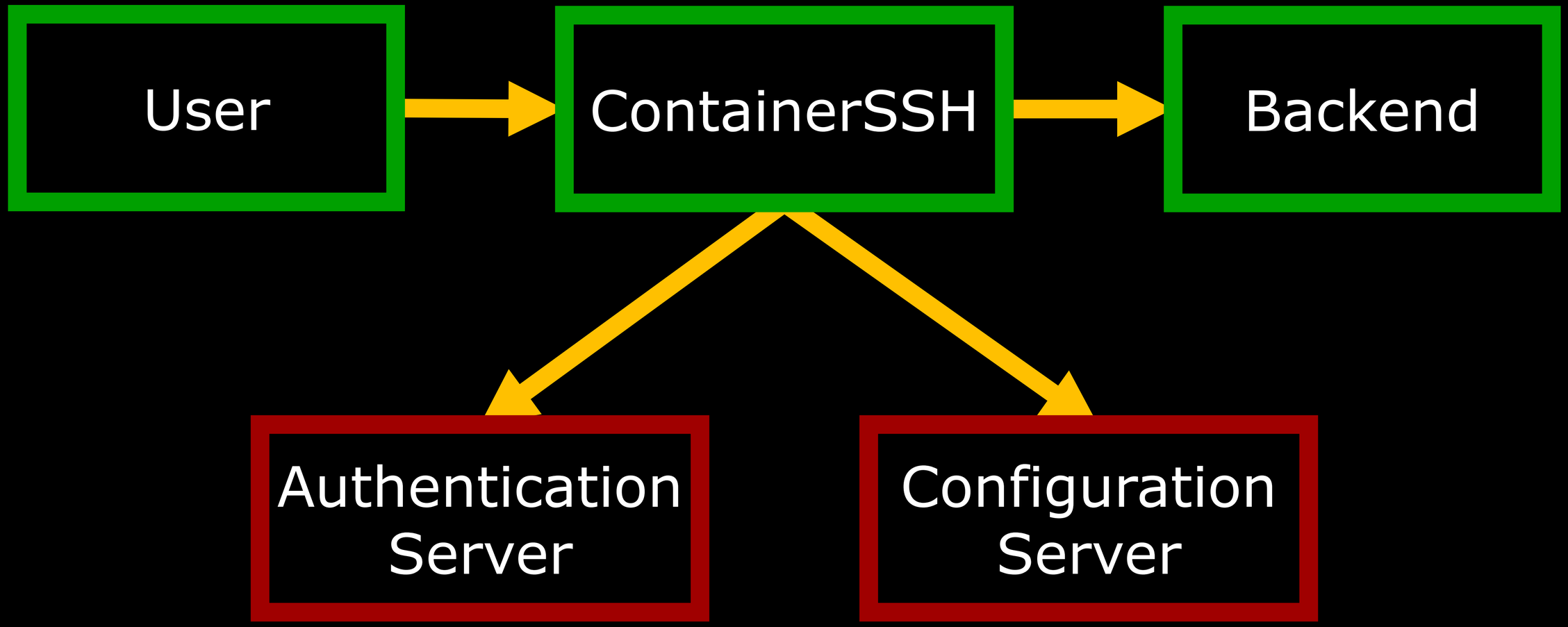




ContainerSSH

Launch containers on demand

containerssh.io


User
ContainerSSH
Backend
Authentication Server
Configuration Server

containerssh.yaml

```
audit:
  format: asciinema
  storage: file
  file:
    directory: /tmp
ssh:
  hostkeys:
    - /etc/ssh/ssh_host_rsa_key
auth:
  url: http://your-auth-server/
```

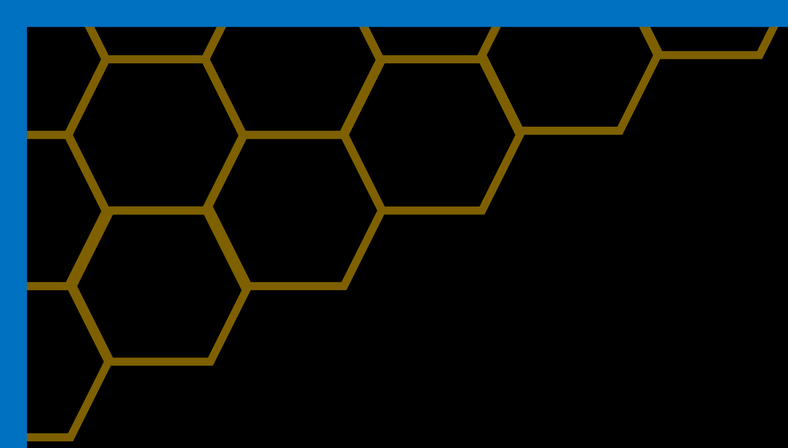
New feature in
ContainerSSH 0.4

```
# ssh-keygen -A
```

```
# docker run -d \
  --restart=always \
  -v /srv/containerssh/config/:/etc/containerssh/ \
  -v /srv/containerssh/audit/:/var/log/containerssh/audit/ \
  --net=host \
  containerssh/containerssh:0.4.0-PR3

# docker run -d \
  --restart=always \
  -p 127.0.0.1:8080:8080 \
  -e CONTAINERSSH_ALLOW_ALL=1 \
  containerssh/containerssh-test-authconfig:0.4.0-PR3
```

```
POST /password HTTP/1.1
Content-Type: application/json

{
  "username": "root",
  "passwordBase64": "...",
  "remoteAddress": "127.0.0.1",
  "connectionId": "..."
}
```

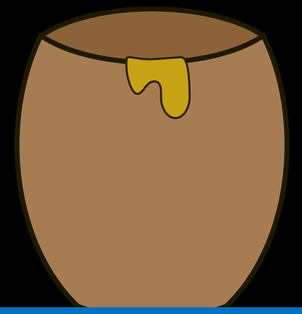
```
HTTP/1.1 200 OK
Content-Type: application/json

{
  "success": true,
}
```



LOGGING

IN BINARY

FORMAT



containerssh.yaml

```
audit:
  format: binary
  storage: file
  file:
    directory: /tmp
ssh:
  hostkeys:
    - /etc/ssh/ssh_host_rsa_key
auth:
  url: http://your-auth-server/
```

{"connectionId":"JUzu5jfs0Fg/4VrP0rhn0A==","timestamp":1604191190206468356,"type":0,"payload":{"remoteAddr":"51.159.64.178"},"ChannelID":-1,"typeName":"auth_connect"}

{"connectionId":"JUzu5jfs0Fg/4VrP0rhn0A==","timestamp":1604191191215197738,"type":100,"payload":{"username":"zxin10","password":null},"ChannelID":-1,"typeName":"auth_password"}

{"connectionId":"JUzu5jfs0Fg/4VrP0rhn0A==","timestamp":1604191191218795395,"type":101,"payload":null,"ChannelID":-1,"typeName":"auth_password_successful"}

{"connectionId":"JUzu5jfs0Fg/4VrP0rhn0A==","timestamp":1604191192366039500,"type":300,"payload":{"channelType":"session"},"ChannelID":-1,"typeName":"new_channel"}

{"connectionId":"JUzu5jfs0Fg/4VrP0rhn0A==","timestamp":1604191192371807824,"type":301,"payload":{"channelType":"session"},"ChannelID":0,"typeName":"new_channel_successful"}

{"connectionId":"JUzu5jfs0Fg/4VrP0rhn0A==","timestamp":1604191192404227483,"type":403,"payload":{"program":"cat /proc/cpuinfo | grep name | wc -l"},"ChannelID":0,"typeName":"exec"}

{"connectionId":"JUzu5jfs0Fg/4VrP0rhn0A==","timestamp":1604191193914837226,"type":500,"payload":{"stream":1,"data":"Mgo="},"ChannelID":0,"typeName":"io"}

{"connectionId":"JUzu5jfs0Fg/4VrP0rhn0A==","timestamp":1604191193972906719,"type":300,"payload":{"channelType":"session"},"ChannelID":-1,"typeName":"new_channel"}

{"connectionId":"JUzu5jfs0Fg/4VrP0rhn0A==","timestamp":1604191193974436934,"type":301,"payload":{"channelType":"session"},"ChannelID":1,"typeName":"new_channel_successful"}

{"connectionId":"JUzu5jfs0Fg/4VrP0rhn0A==","timestamp":1604191194003700157,"type":403,"payload":{"program":"echo -e \"zxin10\\nhHgemapo4oGu\\nhHgemapo4oGu\"|passwd|bash"},"ChannelID":1,"typeName":"exec"}

{"connectionId":"JUzu5jfs0Fg/4VrP0rhn0A==","timestamp":1604191195439900100,"type":500,"payload":{"stream":2,"data":"AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA"},"ChannelID":1,"typeName":"io"}

{"connectionId":"JUzu5jfs0Fg/4VrP0rhn0A==","timestamp":1604191195511429304,"type":300,"payload":{"channelType":"session"},"ChannelID":-1,"typeName":"new_channel"}

{"connectionId":"JUzu5jfs0Fg/4VrP0rhn0A==","timestamp":1604191195512563195,"type":301,"payload":{"channelType":"session"},"ChannelID":2,"typeName":"new_channel_successful"}

{"connectionId":"JUzu5jfs0Fg/4VrP0rhn0A==","timestamp":1604191195544709399,"type":403,"payload":{"program":"echo \"zxin10\\nhHgemapo4oGu\\nhHgemapo4oGu\\n\"|passwd"},"ChannelID":2,"typeName":"exec"}

{"connectionId":"JUzu5jfs0Fg/4VrP0rhn0A==","timestamp":1604191197151404722,"type":500,"payload":{"stream":2,"data":"TmV3IHBhc3N3b3JkOiBSZXR5cGUgbmV3IHBhc3N3b3JkOiBTb3JyeSwgcGFzc3dvcmRzIGRvIG5vdCBtYXRjaC4KcGFzc3dkOiBBdXRoZW50aWNhdGlvbiB0b2tlbiBtYW5pcHVsYXRpb24gZXJyb3IKcGFzc3dkOiBwYXNzd29yZCB1bmNoYW5nZWQK"},"ChannelID":2,"typeName":"io"}

{"connectionId":"JUzu5jfs0Fg/4VrP0rhn0A==","timestamp":1604191197215078761,"type":300,"payload":{"channelType":"session"},"ChannelID":-1,"typeName":"new_channel"}

{"connectionId":"JUzu5jfs0Fg/4VrP0rhn0A==","timestamp":1604191197219040405,"type":301,"payload":{"channelType":"session"},"ChannelID":3,"typeName":"new_channel_successful"}

{"connectionId":"JUzu5jfs0Fg/4VrP0rhn0A==","timestamp":1604191197248343923,"type":403,"payload":{"program":"cat /proc/cpuinfo | grep name | head -n 1 | awk '{print $4,$5,$6,$7,$8,$9;}'"},"ChannelID":3,"typeName":"exec"}

{"connectionId":"JUzu5jfs0Fg/4VrP0rhn0A==","timestamp":1604191198745022396,"type":500,"payload":{"stream":1,"data":"AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA="},"ChannelID":3,"typeName":"io"}

{"connectionId":"JUzu5jfs0Fg/4VrP0rhn0A==","timestamp":1604191198803641665,"type":300,"payload":{"channelType":"session"},"ChannelID":-1,"typeName":"new_channel"}

{"connectionId":"JUzu5jfs0Fg/4VrP0rhn0A==","timestamp":1604191198805206827,"type":301,"payload":{"channelType":"session"},"ChannelID":4,"typeName":"new_channel_successful"}

{"connectionId":"JUzu5jfs0Fg/4VrP0rhn0A==","timestamp":1604191198840593268,"type":403,"payload":{"program":"free -m | grep Mem | awk '{print $2 ,$3, $4, $5, $6, $7}'"},"ChannelID":4,"typeName":"exec"}

{"connectionId":"JUzu5jfs0Fg/4VrP0rhn0A==","timestamp":1604191200586008445,"type":500,"payload":{"stream":1,"data":"MTk4MyAyNDIgMTA4MiAxIDY1OCAxNTkwCg=="},"ChannelID":4,"typeName":"io"}

{program":"cat /proc/cpuinfo | grep name | wc –l"}

{"subsystem":"sftp"}

{"program":"ls -la /dev/ttyGSM* /dev/ttyUSB-mod* /var/spool/sms/* /var/log/smsd.log /etc/smsd.conf* /usr/bin/qmuxd /var/qmux_connect_socket /etc/config/simman /dev/modem* /var/config/sms/*"}

https://gist.github.com/janoszen/4746a93cfdf5203399c699404ea271be

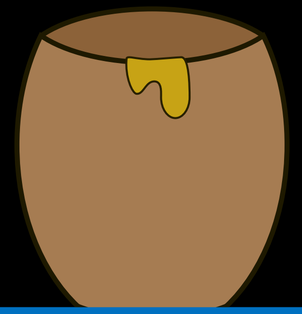
SSH attacks have become sophisticated.

- Change your SSH port.
- Use SSH keys, not passwords.
- Use 4096-bit keys.
- Avoid common usernames.
  (root, test, etc)
- Use key passwords or
  hardware tokens.
- Run security updates weekly.

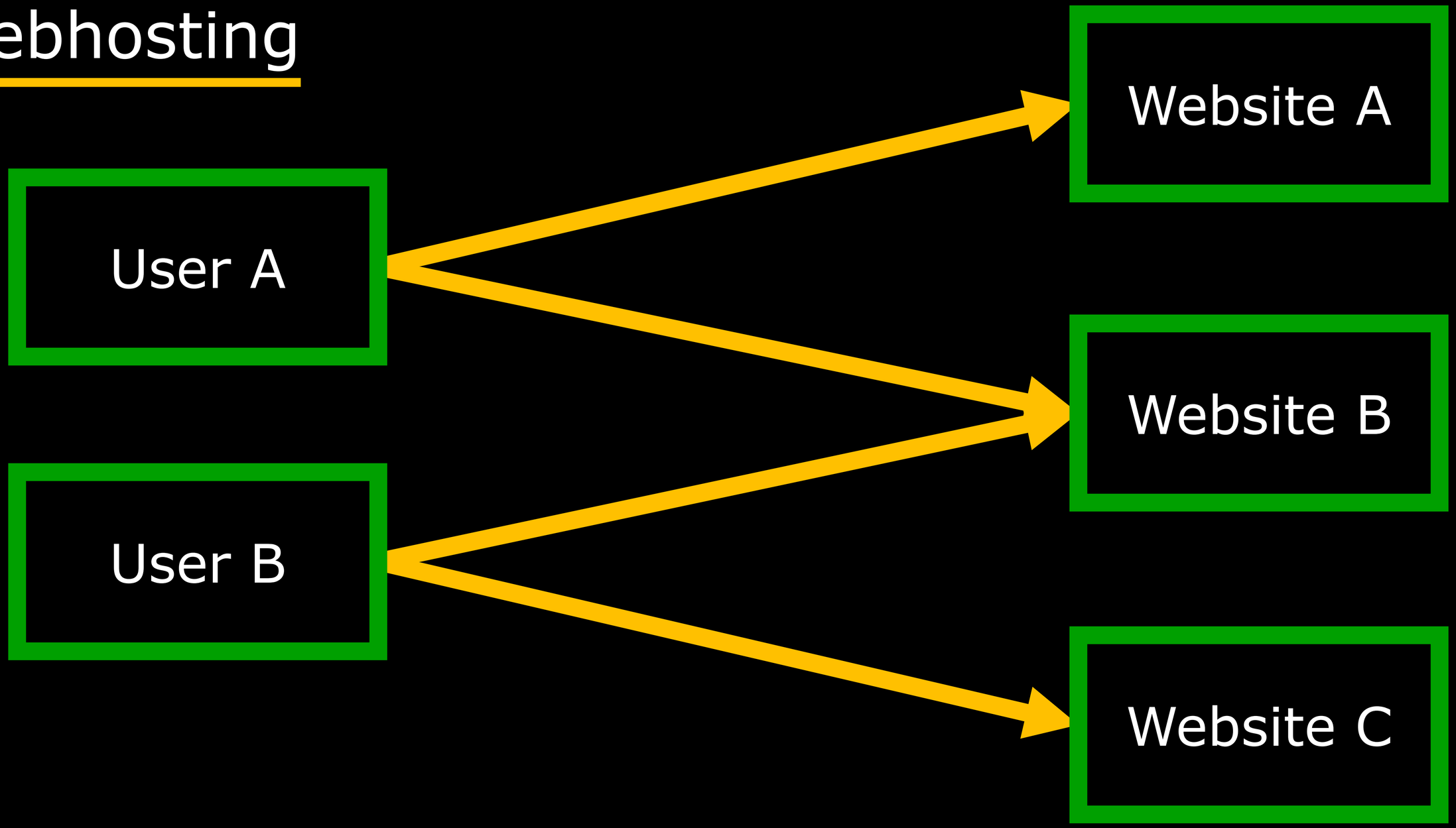
WHAT ELSE

CAN YOU DO

WITH THIS?

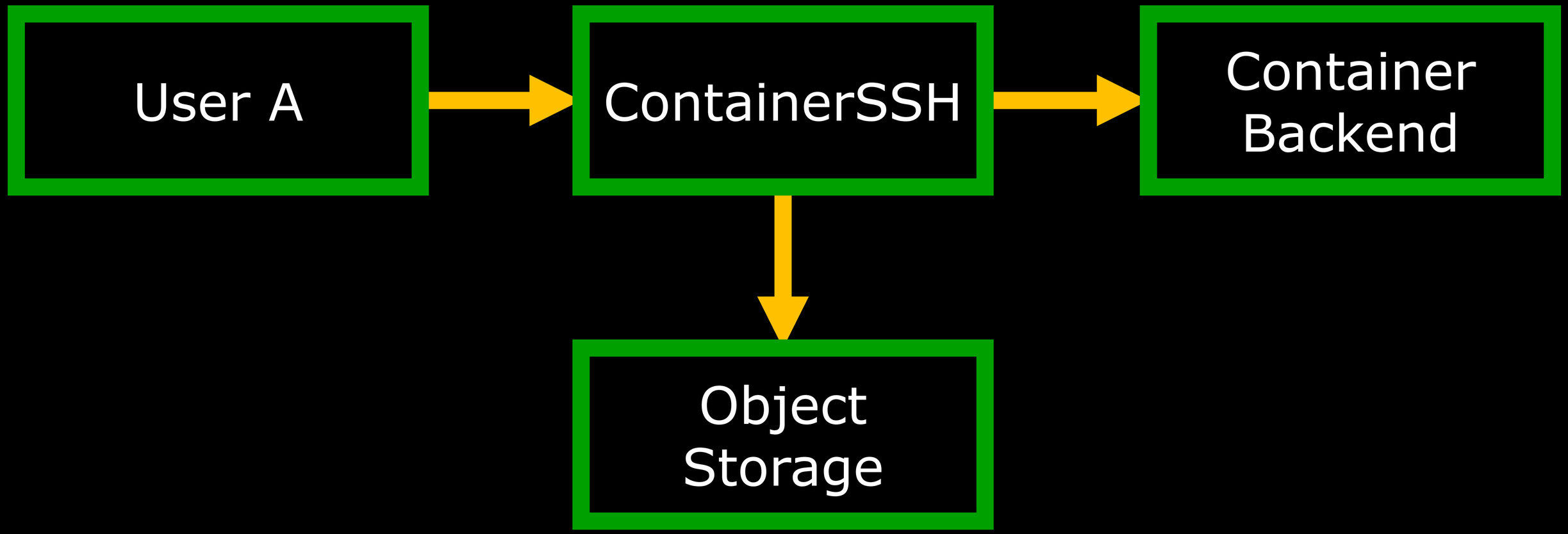


Webhosting



Linux Learning Environment

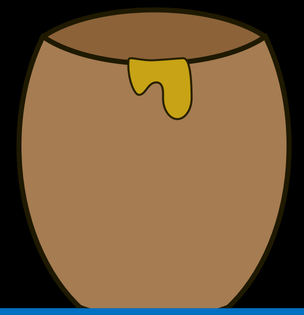
High Security Environment



WHAT DOES

* THE FUTURE *

HOLD?



0.4.0: Audit Logging

This release will feature comprehensive audit logging.

- Add imagePullPolicy to dockerrun backend
- Multi-session containers
- Audit facility
- Integration tests
- Kubernetes backend does not display initial prompt

Future

This release collects all desired, but not scheduled features.

- SSH agent forwarding
- SSH port forwarding
- Keyboard-interactive authentication
- Web client
- Stopping ContainerSSH does not remove containers

Ideas

Ideas that are not sure to come.

- Kerberos/GSS-API authentication
- Direct shell backend
- SSH proxy backend
- PAM authentication
- Support Gitlab CI runners
- File manager

Learn more

containerssh.io

debugged.it