Security (and auditing: code? session)

- Security review: Interest
- How do we know what's going on?
    - Need something like Instrumented SSH, Bro logging
- What are the pieces that should be reviewed to evaluate
- What are the threats or vectors of attack
- How do we secure users from each other
- Security best practices (out of date?)
- Protecting sensitive data
- How do we detect attacks
- What are the biggest gaps?

Ideas

- Security review by someone like Trusted CI
- Interest from Security group at NERSC in a code review
- Document best practice for logging (Hub) and what threats it help address
- Logging syscalls at the OS level
    - File access
    - Network connections
    - Minimal set?
- Identify and document best practices for securing JupyterHub and JupyterLab
- Use network isolation if feasible, could this be done in HPC
- Monitor all opens, writes, network access?
- Use containers to limit exposure and access
    - Plus things like Falco
- Gap: *Comprehensive* Security Guide
    - There is a Jupyter Security mailing list and individual JH, JL, etc.. security pages
- Jupyter Community Workshop on Security?
    - Needs a coordinator, it's a juicy problem, names to be made
    - Someone needs to take the reins here
    - Preparing or including a review
    - Including a variety of labs/industry (and their security people) plus Jupyter project people
- Concept of gradation of security
    - HIPPA vs tutorials vs local notebook
- PR to emit configurable events, plus other efforts happening across Jupyter ecosystem

Known Best Practices

- Log all hub stuff
    - Include authentication and spawners
- Use an external IDP if possible
    - Also: Shane's SSH Auth API is an IDP, admit it Shane
- End-to-end SSL enabled spawners

- Use IPC if possible from the Hub side
    - Document broken kernels
- Limit external access except in container env (?)
- Use network isolation if feasible (VLANs, kube) if possible