THE Ames Laboratory
Creating Materials & Energy Solutions

IOWA STATE UNIVERSITY

DOE National Labs DNSsec Rollout Experiences

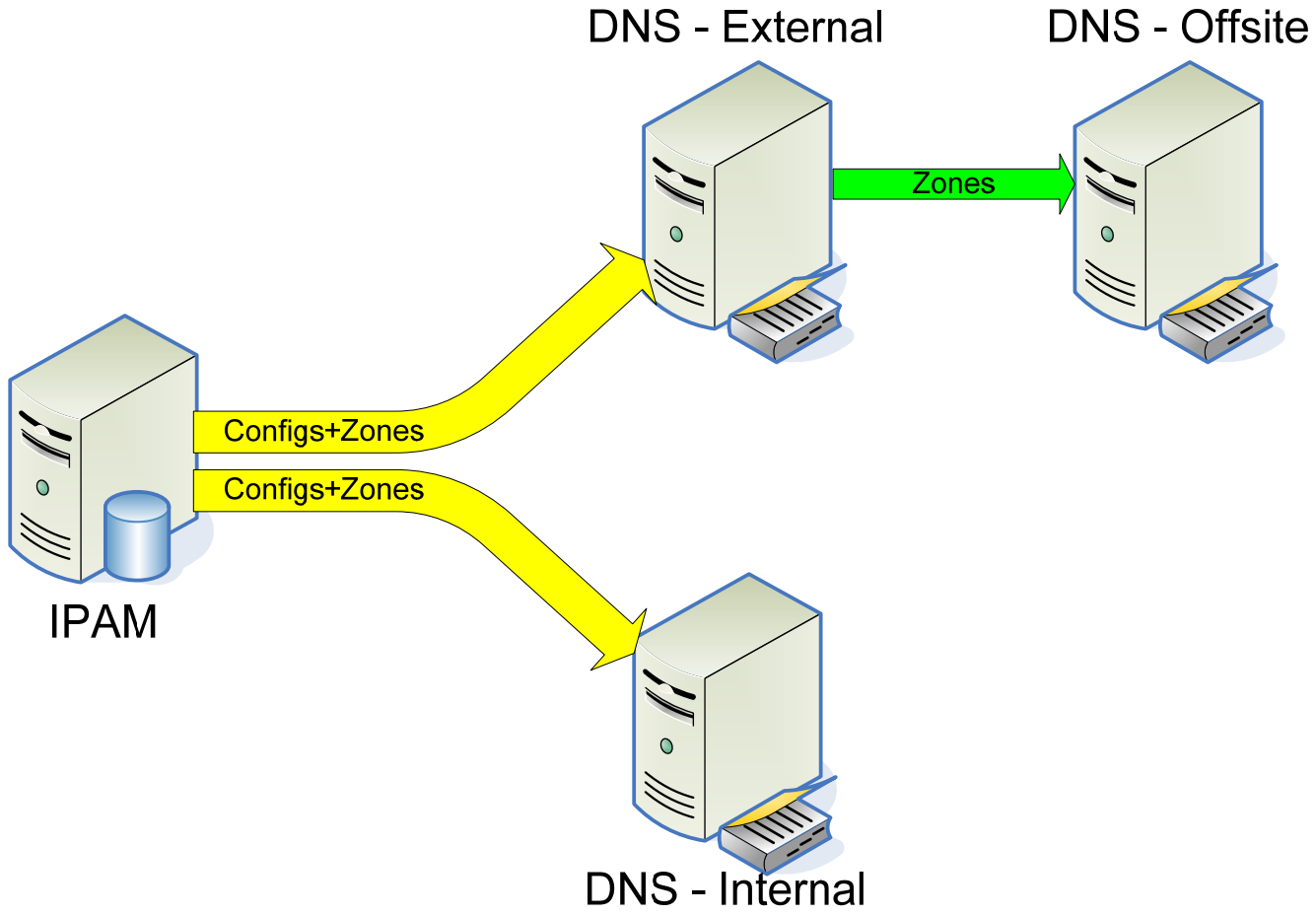
Ames Laboratory Experiences

February 2, 2010
Douglas Stephens, Network/DNS Admin
Ames Laboratory Information Systems
stephens@ameslab.gov

Starting Environment

- BIND 9.6.1 Split-DNS Configuration
  - Internal, External + Offsite slaves with each ISP
  - Split per-zone
- Central Web/SQL IPAM generates
  - BIND configs
  - BIND zone files
  - Pure incrementing zone serials
- Lots of IPAM-Cybersecurity tie-ins
- No DDNS – All zones signed whole
- Forward zones to sign
  - For OMB mandate: 11
  - For FISMA requirement (includes internals & AD): 26

Starting Configuration



Options Explored

- Commercial signing solutions
  - Looked at three: Secure64, Xelerance, Infoblox
  - All were appliance hardware
- Free Open Source (FOSS) signing tools
  - Looked at six
  - Most built around BIND for signing & keygen
- HSMs and other hardware key storage
  - Under no current requirement for FIPS-140 certified solutions

Eval Results: Solution Models

- Bump-in-the-Wire (BitW)
  - In/out data transfer typically uses DNS zone transfer (AXFR)
  - Two commercial solutions, but no FOSS tools found using this model
- IPAM-Centric
  - Signing takes place on or near the site's IPAM system
  - Commercial solutions needed to replace our IPAM to work
  - Most FOSS tools seemed built around this model
- Outsource
  - Work with ISP or other third-party DNS service provider to do signing

Eval Results: NSEC3

- Commercial solutions we looked at could do it
- Many FOSS signing solutions choked
  - Usually unmaintained code or buggy dependent libraries
- Downstream DNS servers carrying NSEC3 zones must also be NSEC3-capable

Eval Results: FOSS Tool Issues

- Package dependency purgatory
- Main issues
  - Zone directory structure assumptions
  - Chroot environment
  - File/directory structures incompatible with our IPAM
  - Buggy key rollovers

Eval Results: Zone Serial Handling

- Problem when zone re-signing triggered from two sources
  - IPAM update with purely incrementing serials.
  - Timed re-sign due to expiring signatures.
- If zone is re-signed due to sig expiration, IPAM could generate a zone serial smaller than published serial
- Solutions
  - Track IPAM and signed serials independently (*)
  - Use datestamp-based zone serials from IPAM and re-sign

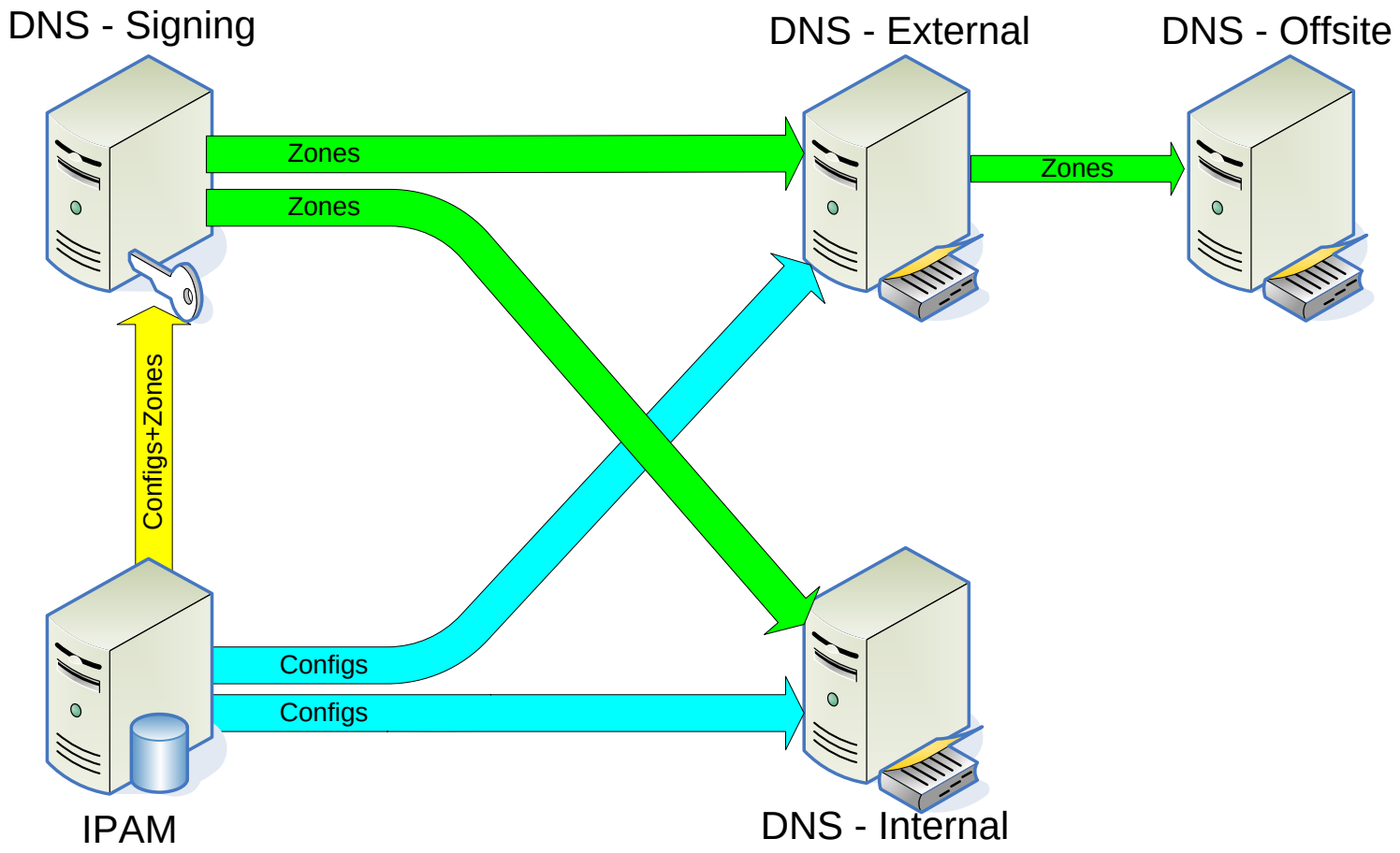
Decision Issues

- Unable to justify cost for commercial BitW solutions
- Not prepared to replace existing IPAM with commercial IPAM-centric solution
  - Too many tie-ins and customizations
- Concerned about commercial IPAM-centric solution availability in time
- Problems with FOSS zone serial incrementing during zone re-sign
- Not prepared to retrofit IPAM to generate date-based zone serials

Current Solution

- Zone and config publishing reconfigured to use signing box as hidden master
- Custom BitW solution
  - SPARTA dnssec-tools `zonesigner` for signing and key generation
  - Custom-written Perl tool
    - Manage key generation and rollover
    - Manage zone/config input from IPAM
    - Trigger re-signing upon IPAM update,zone sig expiration, and key rollover.
    - KSK stage2 roll upon detect new DS RR availability at parent nameservers

Current Configuration



Deployment Status

- Testing custom code for extended-run issues
- Working with one offsite slave running older non-NSEC3-capable BIND
- Planning to incorporate all internal forward zones and AD delegated zones to meet FISMA

End

- Questions?