A Pragmatic Approach For Securing Systems to Process Big Data

Thomas H. Hinke, Ph.D, CISSP
High End Computing Capabilities Project Security Lead
NASA Advanced Supercomputing Division
NASA Ames Research Center
Moffett Field, CA

Target Environment

- Big data and big compute in one system
  - Supports multiple organizations and users
  - Users access system over network such as Internet
  - Users may import and run their own codes
- Data (software, input, output, and other files) has access restrictions
  - Not necessarily accessible to all users
  - May be sensitive
- Data is Moderate or High Sensitivity, but not classified National Defense information
  - As described in Federal Information Processing Standards Publication (FIPS) 199
  - **Moderate Data**: Loss of confidentiality, integrity or availability will have serious adverse effect on organization
  - **High Data**: Loss of confidentiality, integrity or availability will have a severe or catastrophic adverse effect on organization

Example: Supercomputing Center

Main Security Challenges

- System is accessible over the Internet
- Legitimate users must be identified and authenticated
- Users can download their own programs and any additional data that is needed
- Security should not be so onerous that it impedes the use of the system
- Actionable attempts to misuse the system should be identified and acted upon

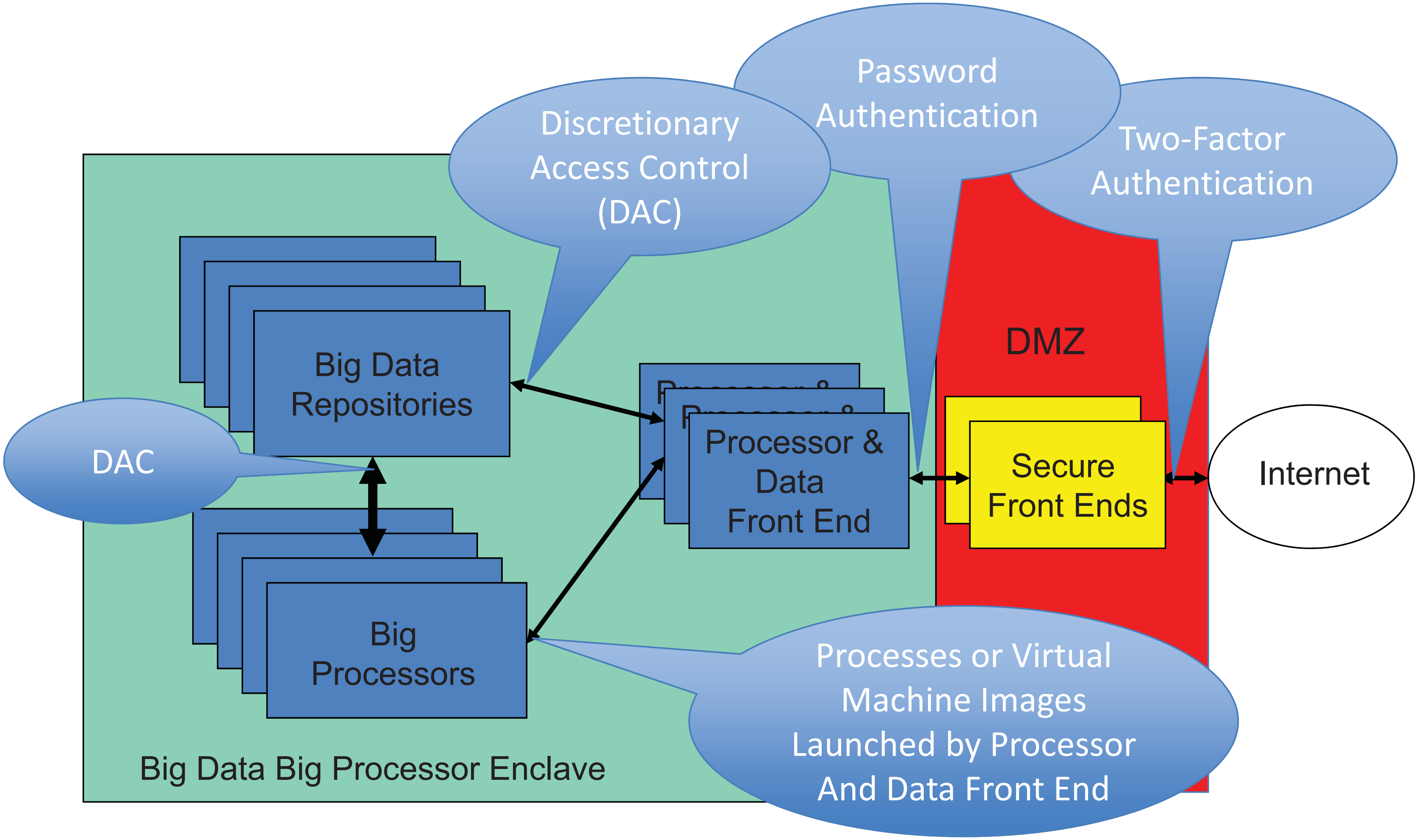
Security Objectives

- Identify and authenticate legitimate users prior to granting them access to the Enclave
- Ensure that users access only data to which they are authorized
- Identify attacks and block them while minimizing
  - False positives – so that legitimate users are not blocked
  - False negatives – so that dangerous attacks are not missed
- Ensure that backdoors are not planted in system

Pragmatic Solution Will Involve

- A simple architecture for protecting the big data and big processing systems
- Minimizing the exposed surface that can be attacked
- Security situational awareness to identify actionable security events

A Pragmatic Architecture for Protecting Big Data and Big Processing Systems


Password Authentication
Discretionary Access Control (DAC)
Two-Factor Authentication
DMZ
Big Data Repositories
DAC
Processor &
Processor &
Processor & Data Front End
Secure Front Ends
Internet
Big Processors
Processes or Virtual Machine Images Launched by Processor And Data Front End
Big Data Big Processor Enclave

Secure Front End Should be Design to be an Attack-Resistant Security Reference Monitor

- Always invoked
- Tamper proof
- Correctly enforces the desired security policy

Secure Front End As Security Reference Monitor Can be Implemented as Follows

Always Invoked Requirement

- Network Access Control Lists (ACLs) ensure that SFEs are only way to access Enclave-resident systems

Tamper Proof Requirement

- Design of the SFE minimizes the opportunity to attack the SFE
- Implemented as a separate device or virtual machine so SFE is isolated from tampering
- Implemented with a jailed (chrooted) environment for all users which limits
  - User access to system directories
  - User access to only those required functions to log in and perform file transfers
- Uses Linux distribution that
  - Allows developer to start with a minimal capability system and
  - Then adds only those additional capabilities that are needed
  - Minimizes the possibility of including unneeded capabilities with potential security vulnerabilities

Correctly Enforces the Desired Security Policy Requirement

- Authenticates users using two-factor authentication

Secure Front End Services Limited to Authentication and File Transfer

- Access to the SFE should be via SSH or some other encrypted connection protocol

- SFEs should support file transfers using SCP or other appropriate protocol
  - Should have sufficient storage capacity so that files could be copied into the SFE
  - Then transferred from the SFE to an Enclave-resident system
  - This would be a two-stage copy
    - From Home system to SFE
    - From SFE to Processor & Data Front End
- SFEs are not the only approach for file transfers
  - Security policy should allow users to pull data into the Enclave from a Processor & Data Front End
  - Users should also be able to push data out of the Enclave from a Processor & Data Front End

The Exposed Surface That Can Be Attacked Should be limited

- The SFEs should be the only system exposed to direct access from the Internet
  - ACLs on network switches/routers should allow Internet access to only the SFEs
- Users on an Enclave-resident Processor & Data Front End can
  - Transfer results out to external systems or
  - Pull in data or programs from external systems
- All originating inbound access from the Internet should be blocked to all Enclave-resident systems

Security Situational Awareness Should Monitoring and Detect

Objective of Security Situational Awareness System is to protect the Enclave from threats such as

- Policy violations
- Vulnerabilities
- Intrusion attempts

Security Situational Awareness Identifies Actionable Security Intrusion Attempts

I1: Intrusion Detection System (IDS) events should be reported upon for only those that are relevant to

- The organization and
- The organization’s installed systems

I2: Back doors into the Enclave should be identified since these could be used

- By Advanced persistent threats (APTs) or others
- To gain and maintain unauthorized access to Enclave systems

I3: APTs and Trojan horse code that is resident in Enclave systems should

- Be identified
- Have any attempt to communicate with external, hostile controllers blocked

Security Situational Awareness Identifies Actionable Security Vulnerabilities

V1: Misconfigurations that could open Enclave up to a successful attack should be identified

- Systems or
- Access Control Lists

V2: Changes in system vulnerabilities that could open Enclave up to successful attack should be identified

V3: New systems that have been attached to the Enclave network should be identified

- If they are previously unknown
- If they have not been checked for security compliance

Security Situational Awareness Identifies Actionable Security Policy Violations

P1: System Administrators should be held to auditable actions

- Identified if they accessing systems directly as root
  - Since root actions do not provide audit trail of who actually performed it

- Required to access systems using SU or SUDO,
  - Since these provide audit trail of who actually performed the action

P2: The sharing of accounts should be identified, since this may indicate

- Unauthorized access or
- Unauthorized sharing of access credentials

P3: Long duration logins should be identified, since logged-in system could provide an entry point into the Enclave for a malicious agent

Questions?