iRODS Policies

integrated Rule Oriented Data System

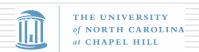
Reagan Moore

{moore, sekar, mwan, schroeder, bzhu, ptooby, antoine, sheauc}@diceresearch.org

{chienyi, marciano, michael_conway}@email.unc.edu













Computer Actionable Rules

- A policy implements a computer actionable rule
 - The rule is composed from micro-services
- Policies are stored in a rule base
 - irods/server/config/reConfigs/core.irb
- The core.irb file can be dynamically changed
 - Lists default rules for all policy-governed actions
 - First valid rules that are found from a top down search will be executed











iRODS Policy Hooks (64)

- Can list the default policies by executing
 - irule -F showcore.ir

```
1051 core.acPreProcForDeleteResource(*RescName)
{ nop }
1052 core.acPostProcForDeleteResource(*RescName)
{ nop }
1053 core.acPreProcForDeleteToken(*TNameSpace,*TName)
{ nop }
1054 core.acPostProcForDeleteToken(*TNameSpace,*TName)
{ nop }
1055 core.acPreProcForModifyResource(*ResourceName,*Option,*NewValue)
{ nop }
```

Most are defaulted to no operation, but can be modified to insert your policy











Implications

- A separate rule engine is installed at each storage location
- A separate rule base is located at each storage location
- To ensure uniform policies, need to update each rule base to contain the same policies
 - Can add policies to the rule base that are unique to the storage system











Rule Syntax

Rules written as

Action-name | Condition | Workflow-chain | Recovery-chain

Action-Name Linked to a specific hook in iRODS

framework, executed each time that

hook is reached

Condition Criteria that must be met for the rule to

execute

Workflow-chain Chain of micro-services and rules that

are executed

Recovery-chain Chain of recovery micro-services that are

invoked on a failure











Rule.ir File

Consists of three lines:

1st line

Rule that is being applied

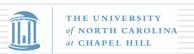
2nd line

Input parameters

3rd line

Output parameters











Interactive rule to list the core.irb file

List of showcore.ir file:

```
myTest||msiAdmShowIRB(*A)|nop
null
*A%ruleExecOut
```

- Very hard to debug for multi-step rules
- Rulegen program provides a more easily debugged syntax
- Rulegen -s showcore.r > showcore.ir











Rulegen Syntax

One micro-service listed per line

```
myTest
{
    msiAdmShowIRB(*A);
}
INPUT *A=null
OUTPUT *A,ruleExecOut
```













Improvements to Rule Engine

 Hao Xu is developing an improved parser to track location of errors within a rule

Example of new parser:









