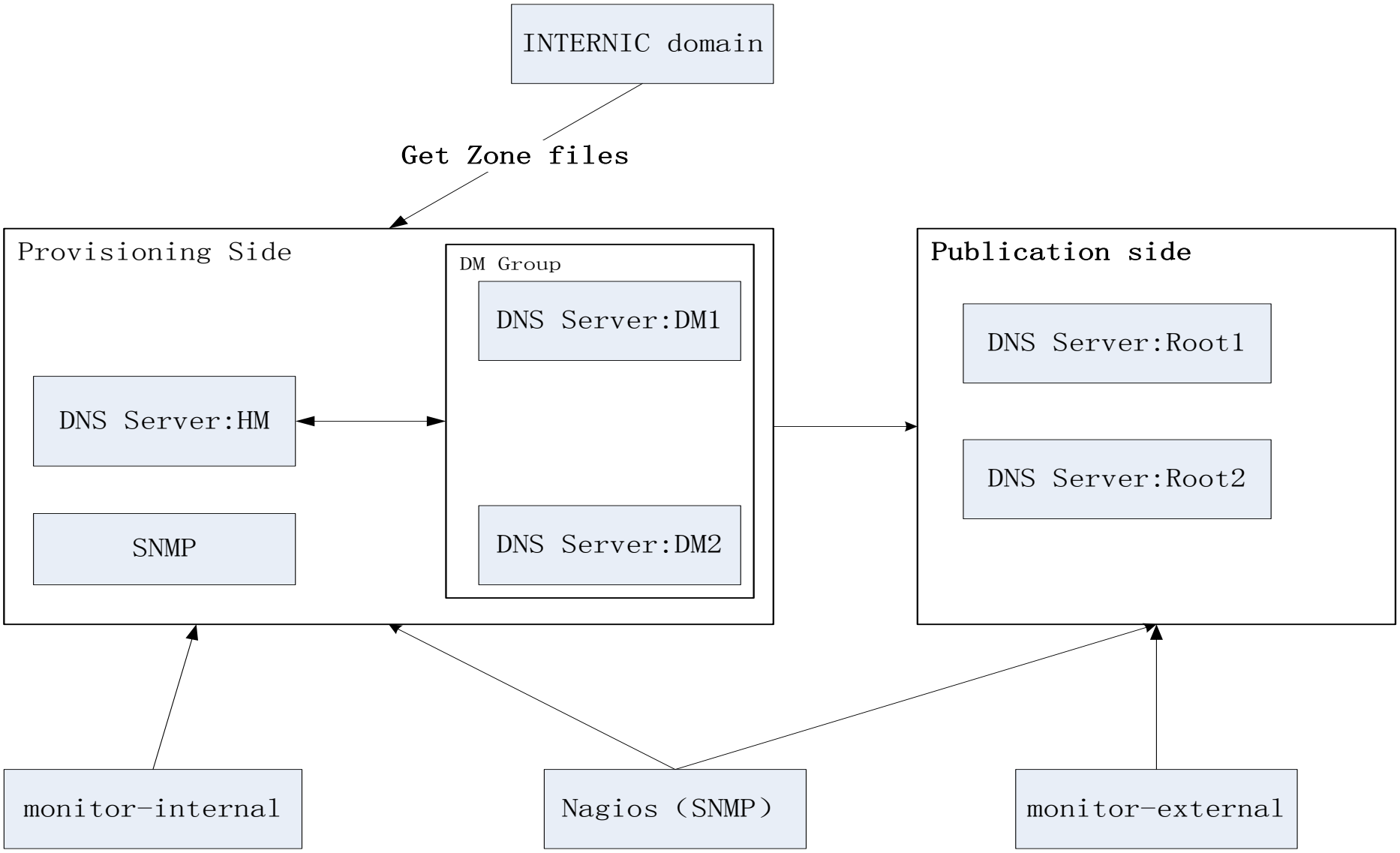
The Monitor System of Macaque Testbed

BII lab

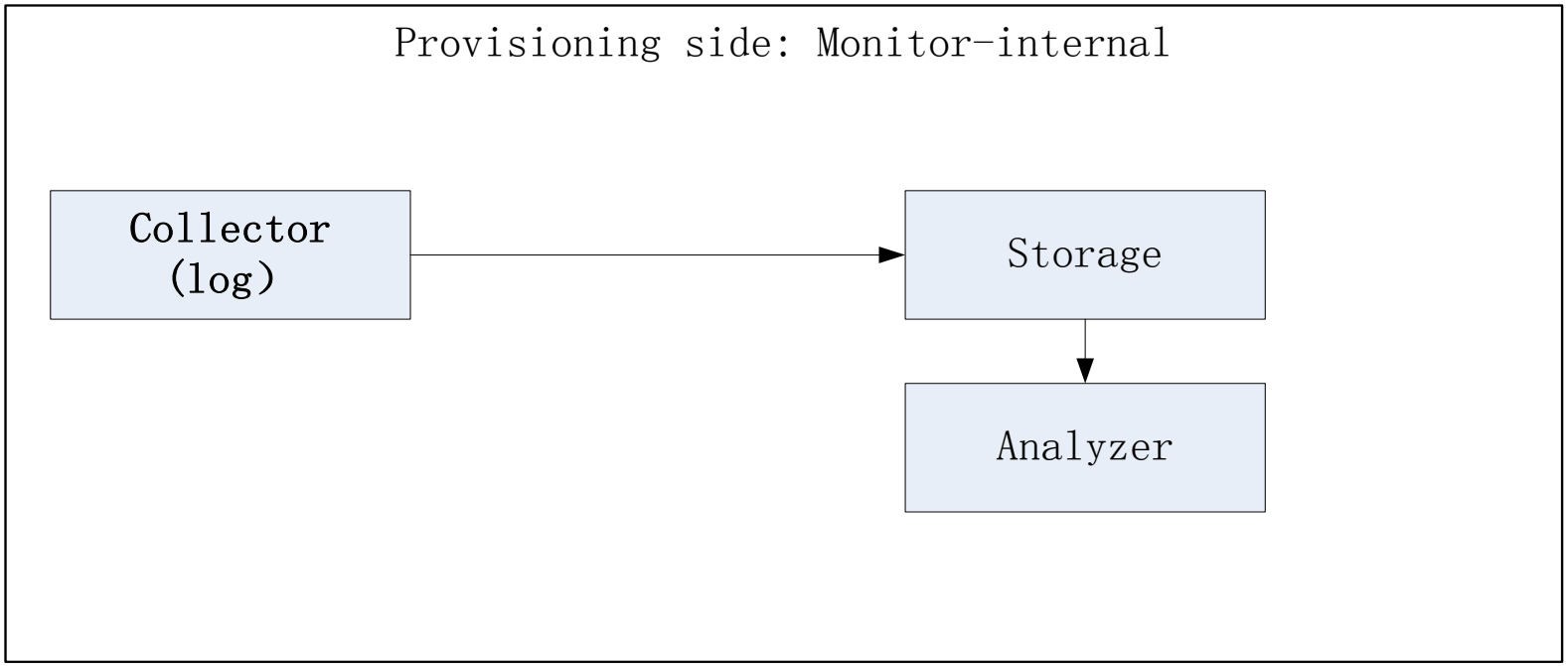
Introduction

- The Monitor System of Macaque Testbed is built to collect all the event and real time data of the test bed for research use of all researchers involved in the project.
- The system includes two sides: Provisioning side and Publication side. Provisioning side will log all event in root system. Publication side will capture all traffic through port 53 in all root server instance.
- BII will be responsible for the storage of data. All the root server is responsible for capture the data.

The architecture of Macaque Project

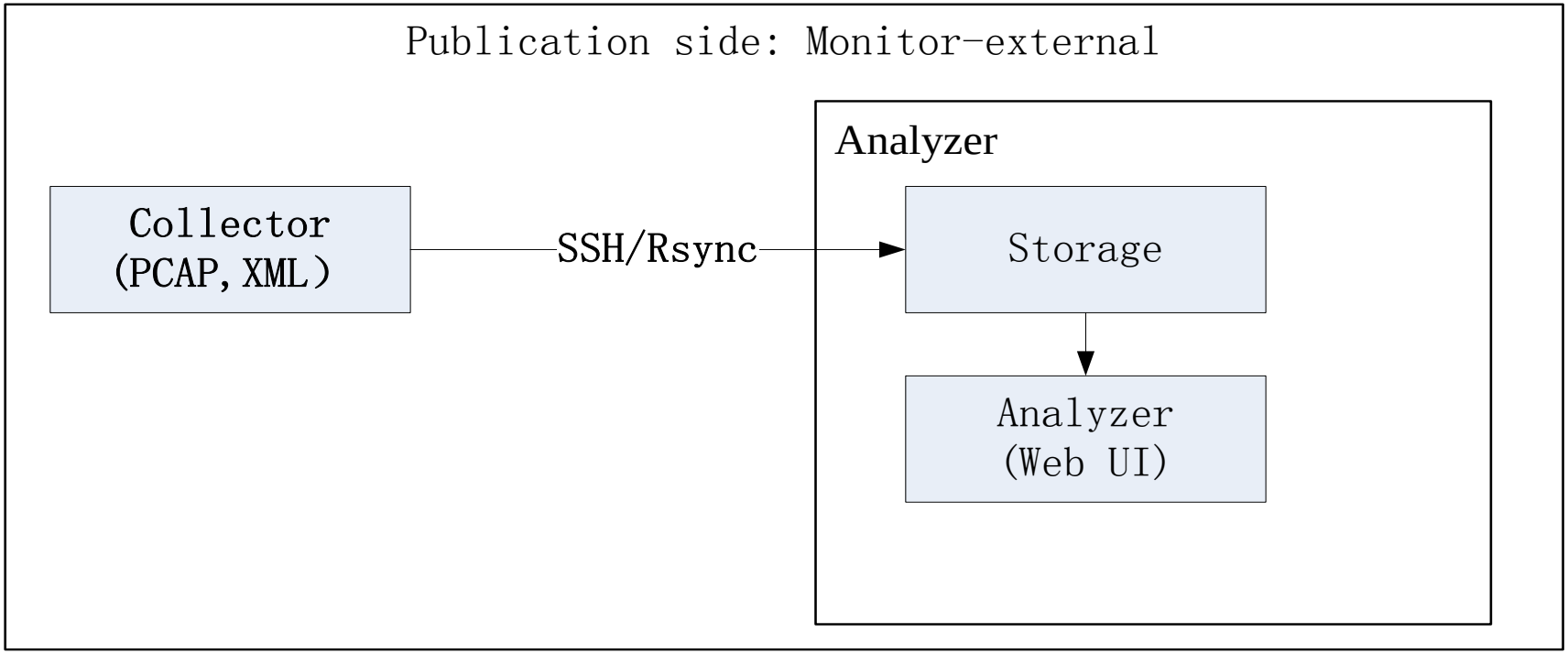


The architecture of Provisioning side



Collector: Using Rsyslogd to collect data.
Storage: Using Mysql as database.
Analyzer: Using scripts to analyze collected data

The architecture of Publication side



Collector: Using PCAP/XML as collector's data type

Storage: Using Mysql as database.

Analyzer: Using Web UI to analyze collected data

Using SSH/Rsync to send data to storage.

Requirement of Publication side data

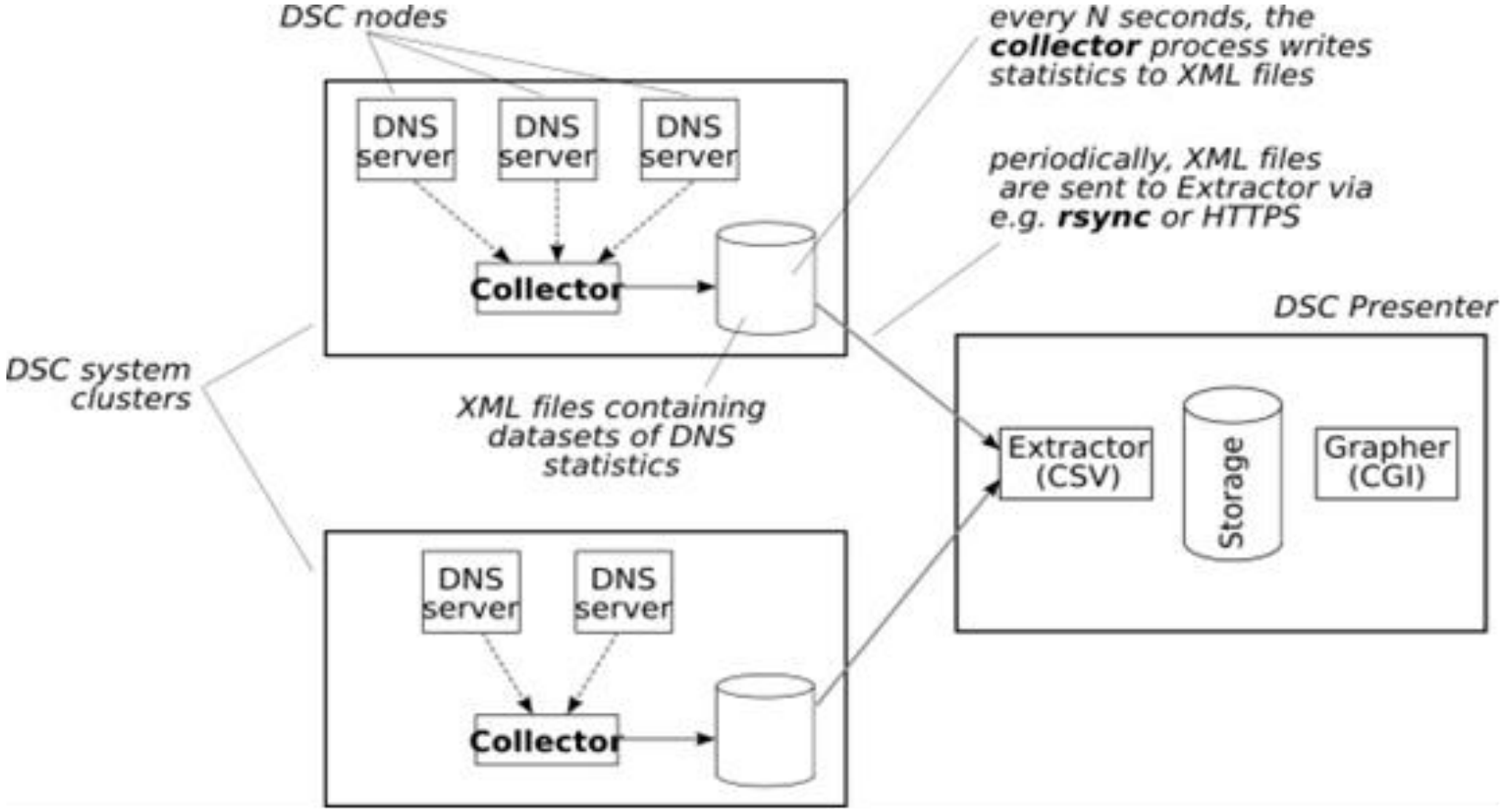
- DITL data: Capture all TCP and UDP data, queries and responses, from all root instances port 53.
- Traffic Data Collector (TDC):Storage format is the pcap file and XML data
  - Dnscap: Capture all dns data on root instance/port mirror.
  - DSC collector: Collect dns data from ethernet interface or pcap file and convert to XML Data

Requirement of Publication side data

- Traffic Data Analyzer (TDA)
  - DSC Presenter: Receives the XML files from collectors. It uses an extractor process to parse and convert them to a different text-based format. The presenter then uses a CGI script to display the data in a Web browser
  - Packetq: PacketQ is a command line tool to run sql queries directly on pcap files. However, PacketQ also contain a very simplistic webserver in order to inspect pcap files remotely and a simple prototype AJAX-based GUI.
  - libpcap/libtrace: TBD

Solutions to Requirement of Publication Side

- DSC as Collector and Analyzer, Monitoring the DNS server in real time.



Solutions to Requirement of Publication Side

- **dnscap + DSC to monitoring the DNS server in real time**
  - dnscap capture all dns data and persistence on storage server
  - Dsc collector convert the pcap file to XML data
  - Rsync the xml data to DSC presenter
- **dnscap + packetq to analyze DNS server with archived dns data at any time.**
  - dnscap capture all dns data and persistence on storage server
  - Packetq run sql queries directly on the pcap files
- libpcap/libtrace: TBD