MIGSOCK

Migratable TCP Socket in Linux

Demonstration of Functionality

Karthik Rajan

Bryan Kuntz

What We Provide

- Socket Migration Capability
  - Linux kernel level implementation
  - Primarily contained within a kernel module; with minor changes to the network TCP files
  - System call API using the ioctl() device interface – allows simple application layer programs to be written which achieve the migration

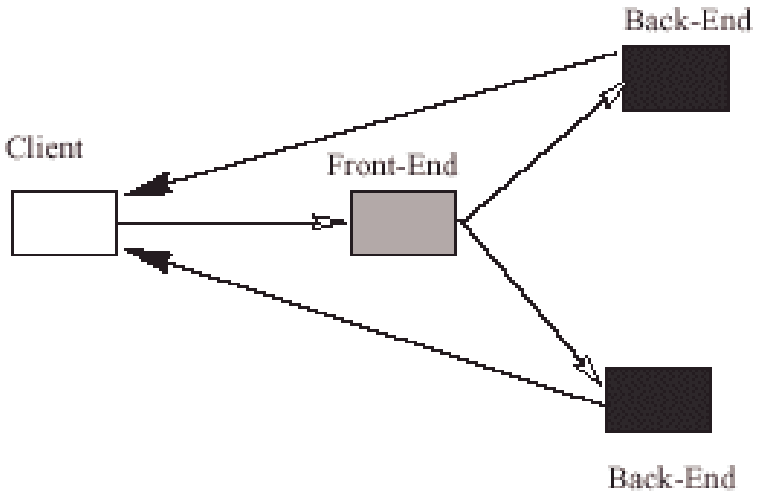
Today

Demonstrate its use with 2 examples:

- Socket handoff
- Socket migrating along with the process

Socket Handoff

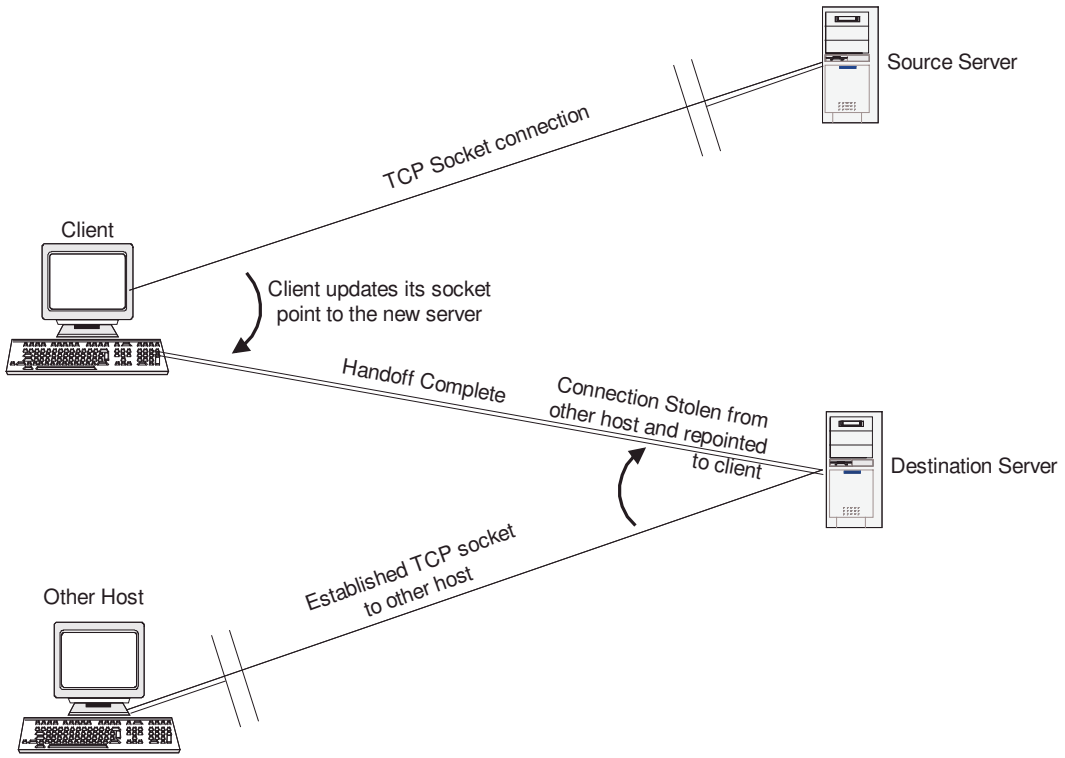
- Migrate only the socket connection; not the process
- Socket communication taken over by another process on the destination host
- Typical use in a Clustered Server environment



Steps

- Suspend an established connection and capture the socket state.
- Transfer this state to a host with a running process.
- This process must be running the same application layer, using a socket descriptor that is connected.
  - In most applications, this will require *STEALING* a socket that is already in the connected state.
- Replace this socket's state with the transferred state.
- Resume the connection with the remote host.

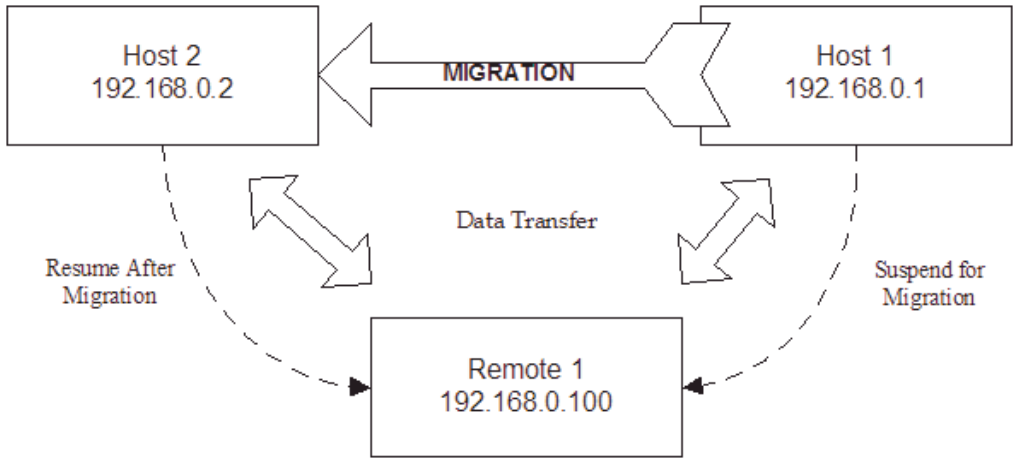
Socket Handoff Scenario



Socket Migration

- A process with an established TCP socket connection migrates to a new host, the socket migrates as well and the connection is maintained.

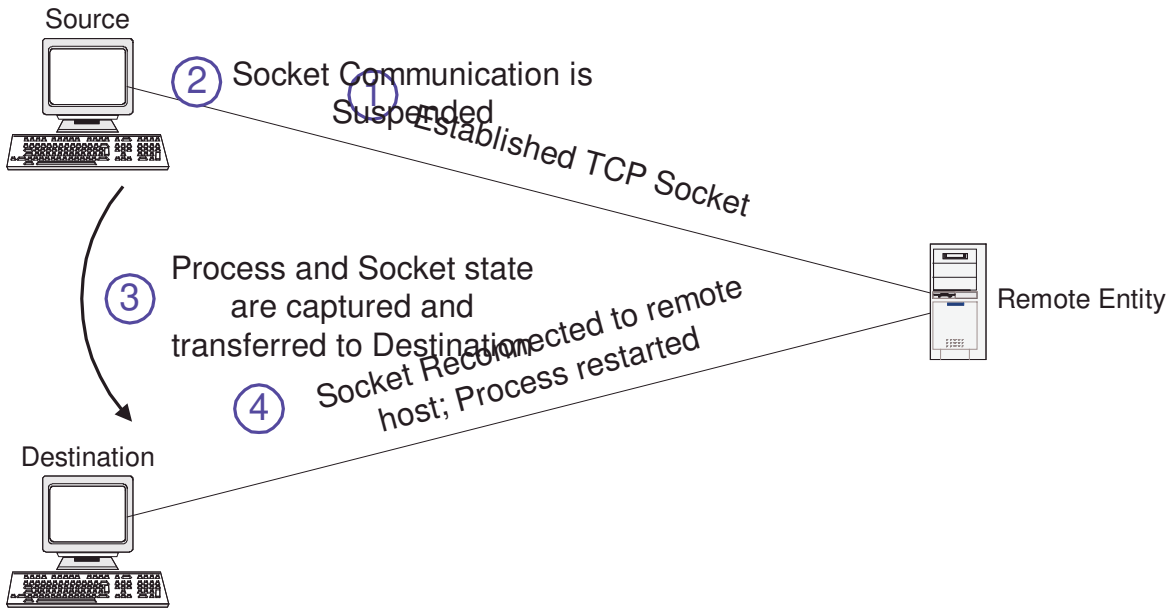
- Requires that an existing process migration mechanism is in place.



Steps

1. Socket migration is initiated at source host
2. TCP message is sent to remote host suspending his side of the connection
3. Socket state is gathered and written to a file
4. State is transferred to the destination host
5. Socket is recreated with new host and port information of the destination host, and connected to remote host
6. TCP message is sent to remote host updating his half of the socket and resuming his side of the connection

Socket Migration Scenario



Questions ?