Mapping XID types to loaded principals

Presenter: Qiaobin Fu

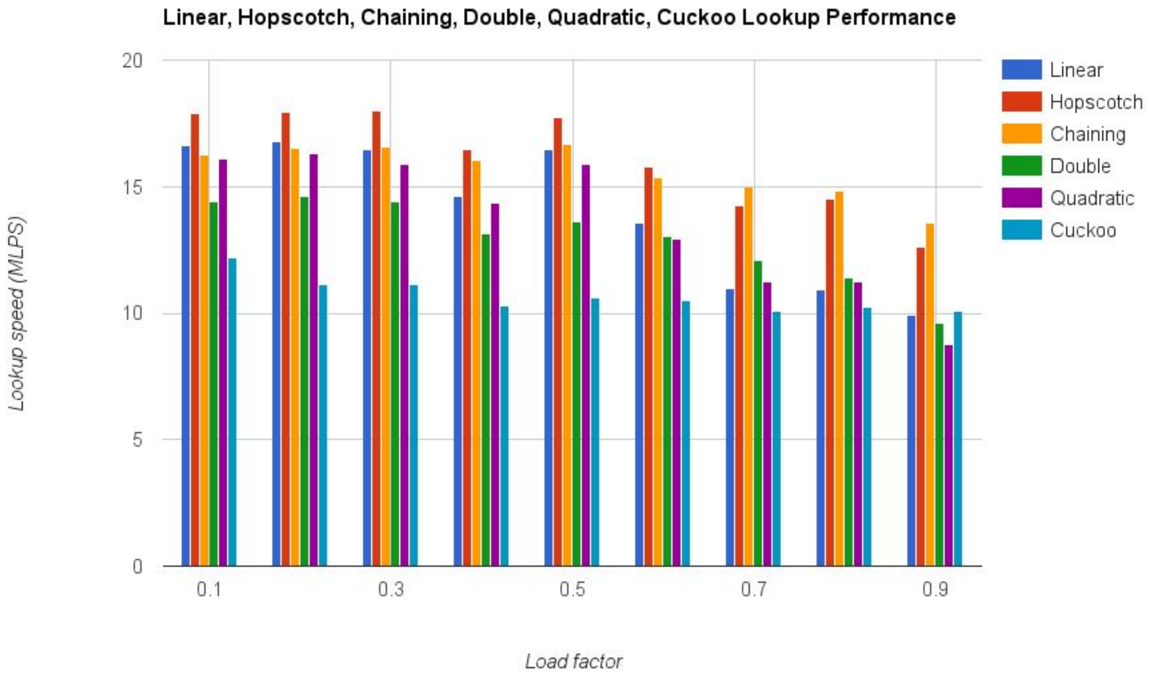
Progress

- ❑ Implemented & ported Hopscotch hashing algorithm
  - ❑ Bugs in the open source code, contacted the author
- ❑ Implemented & ported Linear probing
- ❑ Implemented the Cuckoo hashing algorithm
- ❑ Implemented the Chained hash table
- ❑ Implemented the Double hashing
- ❑ Implemented the Quadratic probing
- ❑ Implemented the Dynamic perfect hashing (Yijun)
- ❑ Implementing the d-left hashing (ongoing - Rishi)

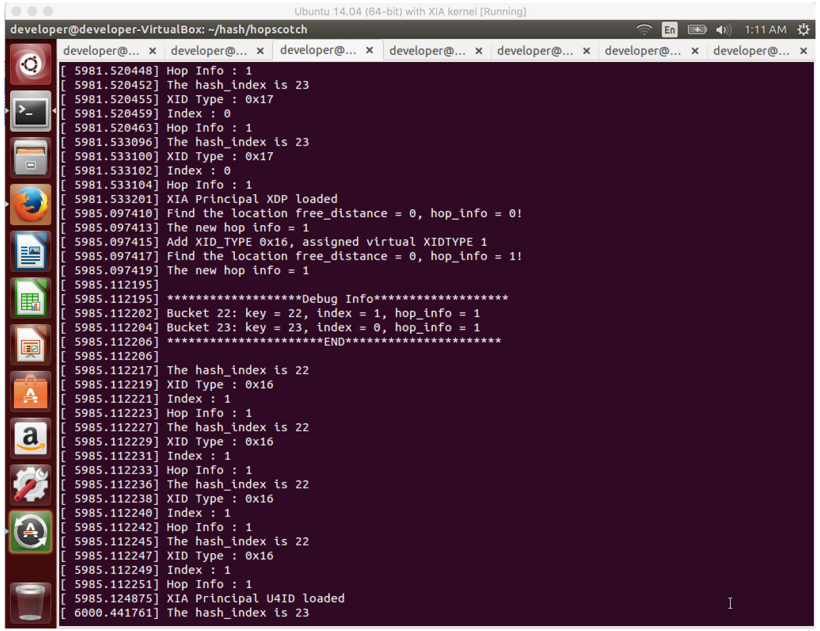
Progress

- Adapt the interface
  - **Did not change the interface at all**
  - **Insertion**: vxt_register_xidty(): called by principals' init function
  - **Deletion**: vxt_unregister_xidty(): called by principals exit function
  - **Lookup**: xt_to_vxt() and xt_to_vxt_rcu()
    - Several functions in source files: flow.c, route.c, and socket.c

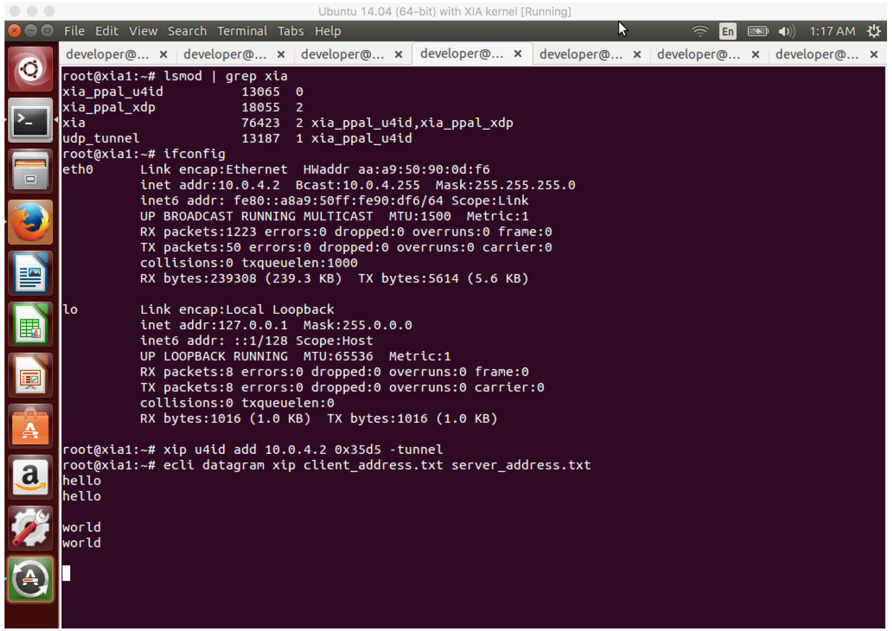
Performance comparison



Experiment on Linux XIA - Hopscotch


Ubuntu 14.04 (64-bit) with XIA kernel [Running]
developer@developer-VirtualBox: ~/hash/hopscotch
1:11 AM

```
[ 5981.520448] Hop Info : 1
[ 5981.520452] The hash_index is 23
[ 5981.520455] XID Type : 0x17
[ 5981.520459] Index : 0
[ 5981.520463] Hop Info : 1
[ 5981.533096] The hash_index is 23
[ 5981.533100] XID Type : 0x17
[ 5981.533102] Index : 0
[ 5981.533104] Hop Info : 1
[ 5981.533201] XIA Principal XDP loaded
[ 5985.097410] Find the location free_distance = 0, hop_info = 0!
[ 5985.097413] The new hop info = 1
[ 5985.097415] Add XID_TYPE 0x16, assigned virtual XIDTYPE 1
[ 5985.097417] Find the location free_distance = 0, hop_info = 1!
[ 5985.097419] The new hop info = 1
[ 5985.112195]
[ 5985.112195] ******************Debug Info******************
[ 5985.112202] Bucket 22: key = 22, index = 1, hop_info = 1
[ 5985.112204] Bucket 23: key = 23, index = 0, hop_info = 1
[ 5985.112206] *********************END**********************
[ 5985.112206]
[ 5985.112217] The hash_index is 22
[ 5985.112219] XID Type : 0x16
[ 5985.112221] Index : 1
[ 5985.112223] Hop Info : 1
[ 5985.112227] The hash_index is 22
[ 5985.112229] XID Type : 0x16
[ 5985.112231] Index : 1
[ 5985.112233] Hop Info : 1
[ 5985.112236] The hash_index is 22
[ 5985.112238] XID Type : 0x16
[ 5985.112240] Index : 1
[ 5985.112242] Hop Info : 1
[ 5985.112245] The hash_index is 22
[ 5985.112247] XID Type : 0x16
[ 5985.112249] Index : 1
[ 5985.112251] Hop Info : 1
[ 5985.124875] XIA Principal U4ID loaded
[ 6000.441761] The hash_index is 23
```


Ubuntu 14.04 (64-bit) with XIA kernel [Running]
File Edit View Search Terminal Tabs Help
1:17 AM

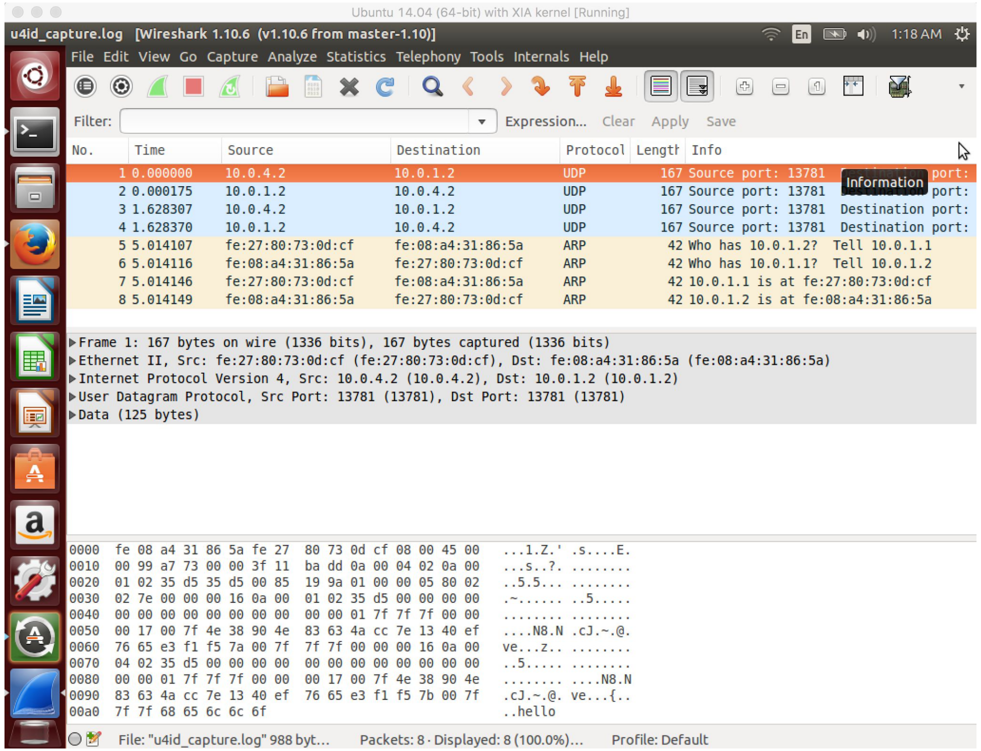
```
root@xia1:~# lsmod | grep xia
xia_ppal_u4id          13065  0
xia_ppal_xdp           18055  2
xia                    76423  2 xia_ppal_u4id,xia_ppal_xdp
udp_tunnel             13187  1 xia_ppal_u4id
root@xia1:~# ifconfig
eth0      Link encap:Ethernet  HWaddr aa:a9:50:90:0d:f6
          inet addr:10.0.4.2  Bcast:10.0.4.255  Mask:255.255.255.0
          inet6 addr: fe80::a8a9:50ff:fe90:df6/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:1223 errors:0 dropped:0 overruns:0 frame:0
          TX packets:50 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:239308 (239.3 KB)  TX bytes:5614 (5.6 KB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:8 errors:0 dropped:0 overruns:0 frame:0
          TX packets:8 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:1016 (1.0 KB)  TX bytes:1016 (1.0 KB)

root@xia1:~# xip u4id add 10.0.4.2 0x35d5 -tunnel
root@xia1:~# ecli datagram xip client_address.txt server_address.txt
hello
hello

world
world
```

Experiment - Hopscotch


Ubuntu 14.04 (64-bit) with XIA kernel [Running]
u4id_capture.log [Wireshark 1.10.6 (v1.10.6 from master-1.10)]
1:18 AM
File Edit View Go Capture Analyze Statistics Telephony Tools Internals Help
Filter:
Expression... Clear Apply Save
Information

```
No.  Time       Source             Destination        Protocol Lengtl Info
   1 0.000000   10.0.4.2           10.0.1.2           UDP      167 Source port: 13781  Destination port:
   2 0.000175   10.0.1.2           10.0.4.2           UDP      167 Source port: 13781  Destination port:
   3 1.628307   10.0.4.2           10.0.1.2           UDP      167 Source port: 13781  Destination port:
   4 1.628370   10.0.1.2           10.0.4.2           UDP      167 Source port: 13781  Destination port:
   5 5.014107   fe:27:80:73:0d:cf  fe:08:a4:31:86:5a  ARP       42 Who has 10.0.1.2?  Tell 10.0.1.1
   6 5.014116   fe:08:a4:31:86:5a  fe:27:80:73:0d:cf  ARP       42 Who has 10.0.1.1?  Tell 10.0.1.2
   7 5.014146   fe:27:80:73:0d:cf  fe:08:a4:31:86:5a  ARP       42 10.0.1.1 is at fe:27:80:73:0d:cf
   8 5.014149   fe:08:a4:31:86:5a  fe:27:80:73:0d:cf  ARP       42 10.0.1.2 is at fe:08:a4:31:86:5a
```

```
▶Frame 1: 167 bytes on wire (1336 bits), 167 bytes captured (1336 bits)
▶Ethernet II, Src: fe:27:80:73:0d:cf (fe:27:80:73:0d:cf), Dst: fe:08:a4:31:86:5a (fe:08:a4:31:86:5a)
▶Internet Protocol Version 4, Src: 10.0.4.2 (10.0.4.2), Dst: 10.0.1.2 (10.0.1.2)
▶User Datagram Protocol, Src Port: 13781 (13781), Dst Port: 13781 (13781)
▶Data (125 bytes)
```

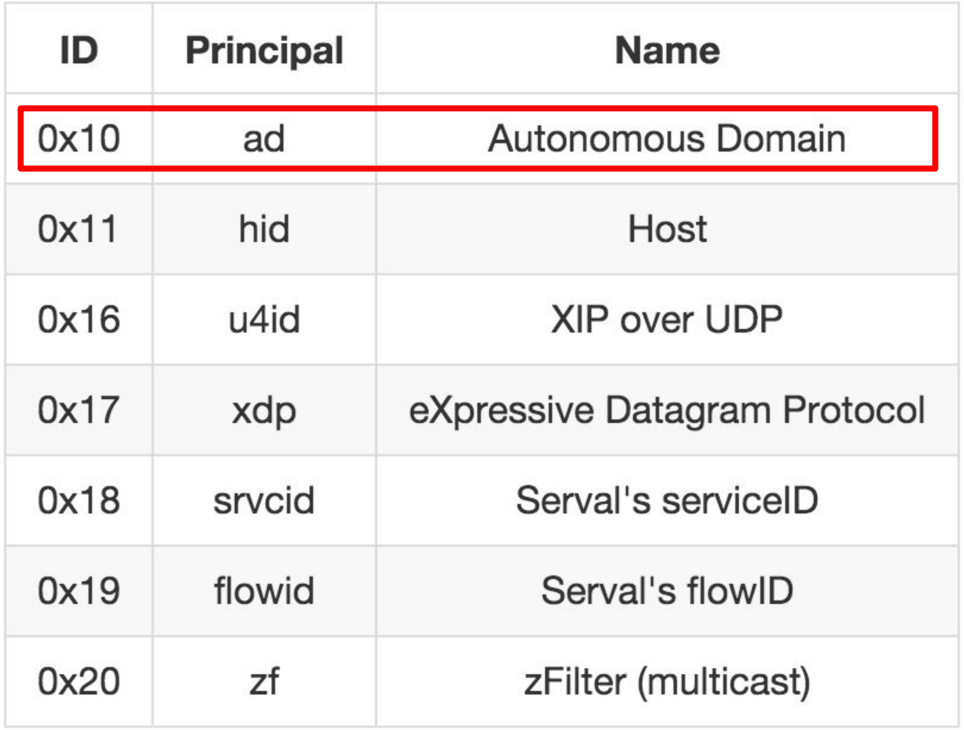
```
0000  fe 08 a4 31 86 5a fe 27  80 73 0d cf 08 00 45 00   ...1.Z.' .s....E.
0010  00 99 a7 73 00 00 3f 11  ba dd 0a 00 04 02 0a 00   ...s..?. ........
0020  01 02 35 d5 35 d5 00 85  19 9a 01 00 00 05 80 02   ..5.5... ........
0030  02 7e 00 00 00 16 0a 00  01 02 35 d5 00 00 00 00   .~...... ..5.....
0040  00 00 00 00 00 00 00 00  00 00 01 7f 7f 7f 00 00   ........ ........
0050  00 17 00 7f 4e 38 90 4e  83 63 4a cc 7e 13 40 ef   ....N8.N .cJ.~.@.
0060  76 65 e3 f1 f5 7a 00 7f  7f 7f 00 00 00 16 0a 00   ve...z.. ........
0070  04 02 35 d5 00 00 00 00  00 00 00 00 00 00 00 00   ..5..... ........
0080  00 00 01 7f 7f 7f 00 00  00 17 00 7f 4e 38 90 4e   ........ ....N8.N
0090  83 63 4a cc 7e 13 40 ef  76 65 e3 f1 f5 7b 00 7f   .cJ.~.@. ve...{..
00a0  7f 7f 68 65 6c 6c 6f                               ..hello
```

File: "u4id_capture.log" 988 byt...   Packets: 8 · Displayed: 8 (100.0%)...   Profile: Default

Work to be done

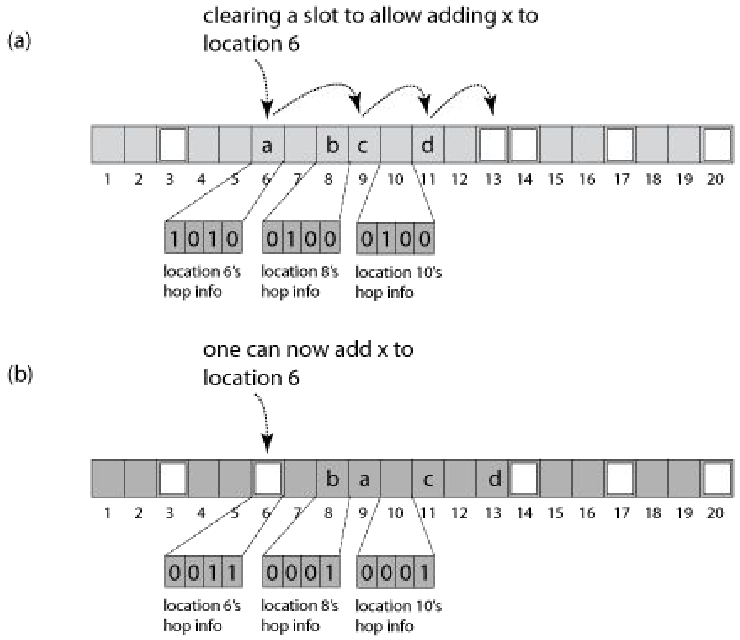
- Conduct more experiments to test
  - Lookup speed
  - Load factor
  - etc.

Principals in Linux XIA



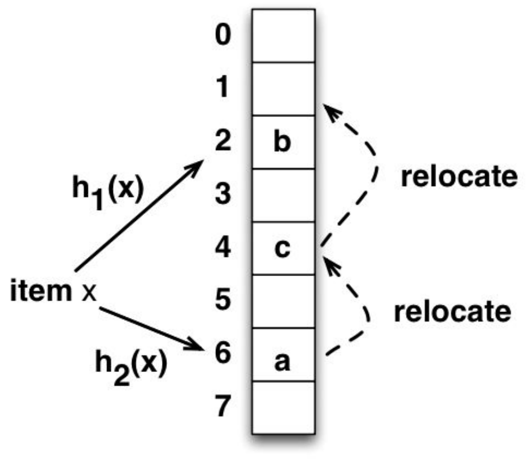
ID	Principal	Name
0x10	ad	Autonomous Domain
0x11	hid	Host
0x16	u4id	XIP over UDP
0x17	xdp	eXpressive Datagram Protocol
0x18	srvcid	Serval's serviceID
0x19	flowid	Serval's flowID
0x20	zf	zFilter (multicast)

Hopscotch hashing

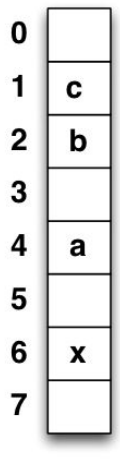


- ❏ A single hash function *h*
- ❏ The item hashed to an entry will always be found either in that entry, or in one of the next H − 1 entries, where H is a constant
- ❏ Each entry includes a hop-information word, an H-bit bitmap that indicates which of the next H−1 entries contain items that hashed to the current entry's virtual bucket

Cuckoo hashing



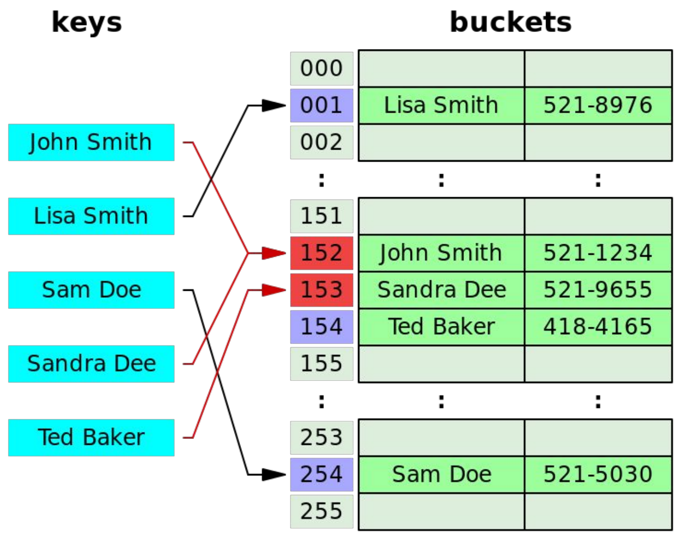
(a) before inserting item x



(b) after item x inserted

- An array of buckets
- Each item has two candidate buckets

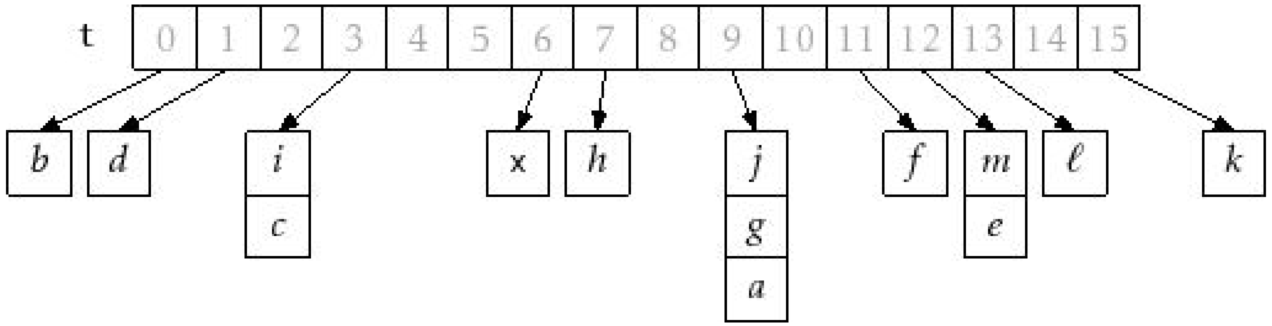
Linear probing



In linear probing, all entry records are stored in the bucket array itself. When a new entry has to be inserted, the buckets are examined, starting with the hashed-to slot and proceeding in some *probe sequence*, until an unoccupied slot is found

- "Ted Baker" has a unique hash,
- But collided with "Sandra Dee", that had previously collided with "John Smith"
- Variants:
  - Quadratic probing
  - Double hashing

Chained hash table



An example of a Chained Hash Table with 14 entries

Reference

1. http://opendatastructures.org/versions/edition-0.1e/ods-java/5_1_ChainedHashTable_Hashin.html
2. https://en.wikipedia.org/wiki/Hash_table#Collision_resolution
3. Herlihy, Maurice, Nir Shavit, and Moran Tzafrir. "Hopscotch hashing." *Distributed Computing*. Springer Berlin Heidelberg, 2008. 350-364.