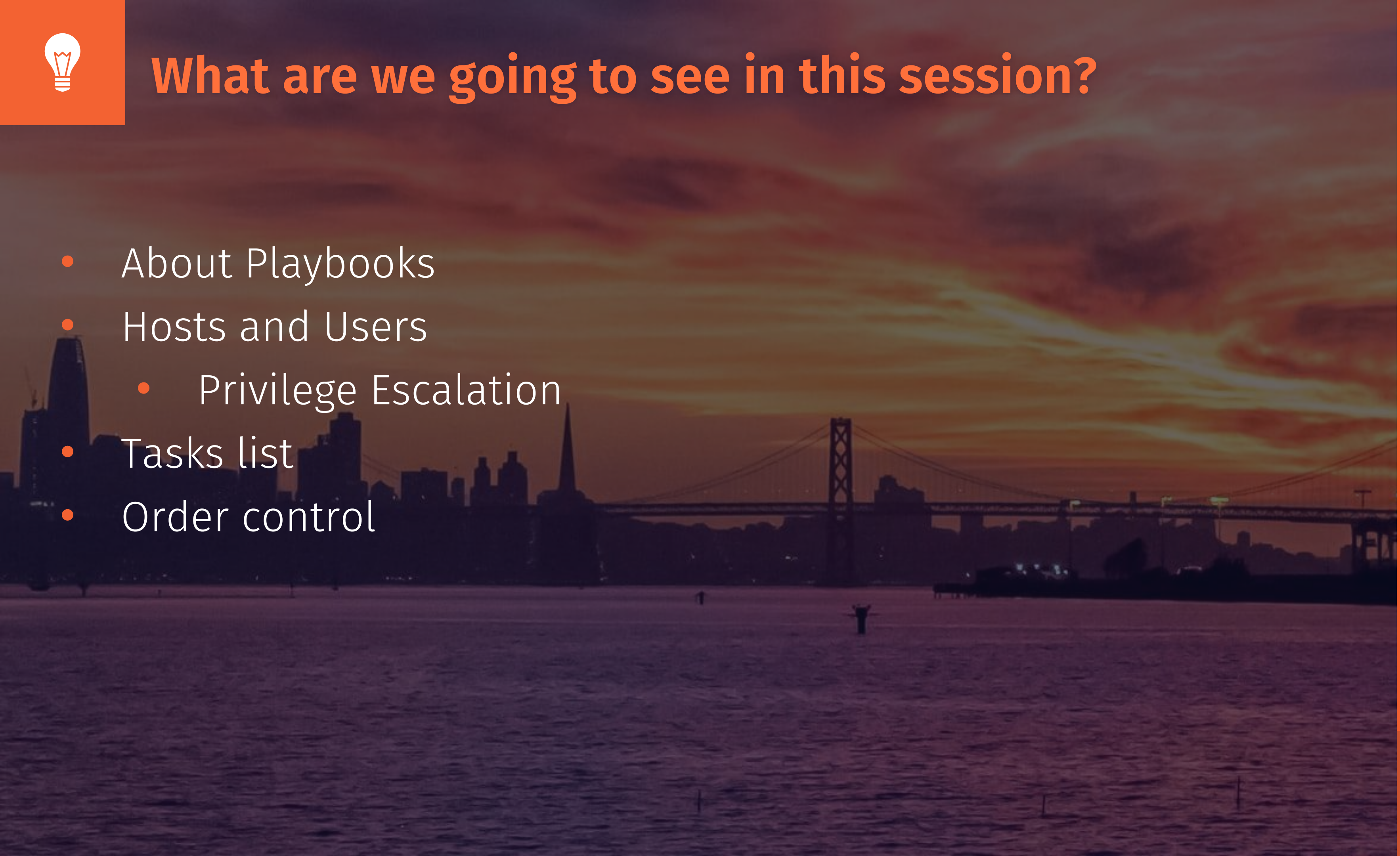
What are we going to see in this session?



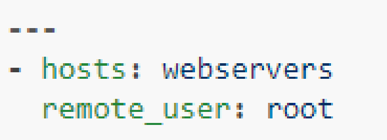
- About Playbooks
- Hosts and Users
  - Privilege Escalation
- Tasks list
- Order control

About Playbooks

- Playbooks are completely different way to use ansible than in ad-hoc execution mode.
- Playbook is very powerful concept in Ansible.
- Playbooks are the basis for real simple configuration management and multi machine deployment.
- Playbooks are helpful and well suited to deploy complex applications.
- Using playbook you can orchestrate steps of any manual process which you do in daily basis.
- They can launch tasks synchronously and asynchronously.
- Playbooks are more likely to be kept in source control and used to push out your configuration to remote machines.

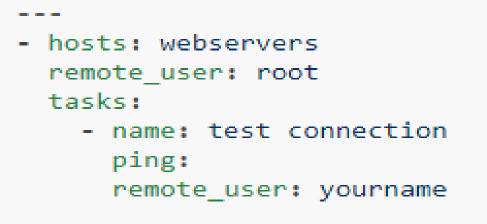
Hosts and Users

- For each play in a playbook, you get to choose which machines in your infrastructure to target and what remote user to complete the tasks.
- The [hosts] line is a list of one or more groups or host patterns, separated by colons.
- The [remote_user] is just the name of the user account.



```
---
- hosts: webservers
  remote_user: root
```

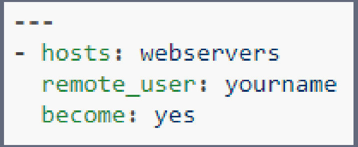
- You can also define remote users per tasks



```
---
- hosts: webservers
  remote_user: root
  tasks:
    - name: test connection
      ping:
      remote_user: yourname
```

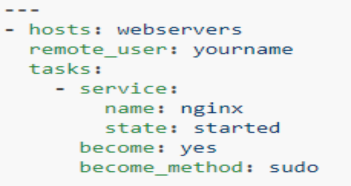
Privilege Escalation

- Ansible supports for running things as another user with Privilege Escalation.



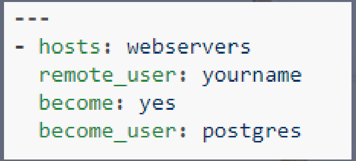
```
---
- hosts: webservers
  remote_user: yourname
  become: yes
```

- You can also use keyword become on a particular task instead of the whole play.



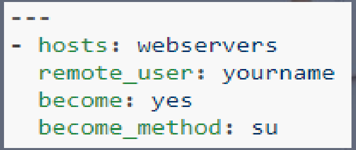
```
---
- hosts: webservers
  remote_user: yourname
  tasks:
    - service:
        name: nginx
        state: started
      become: yes
      become_method: sudo
```

- You can also login as you and then become a different user other then root.



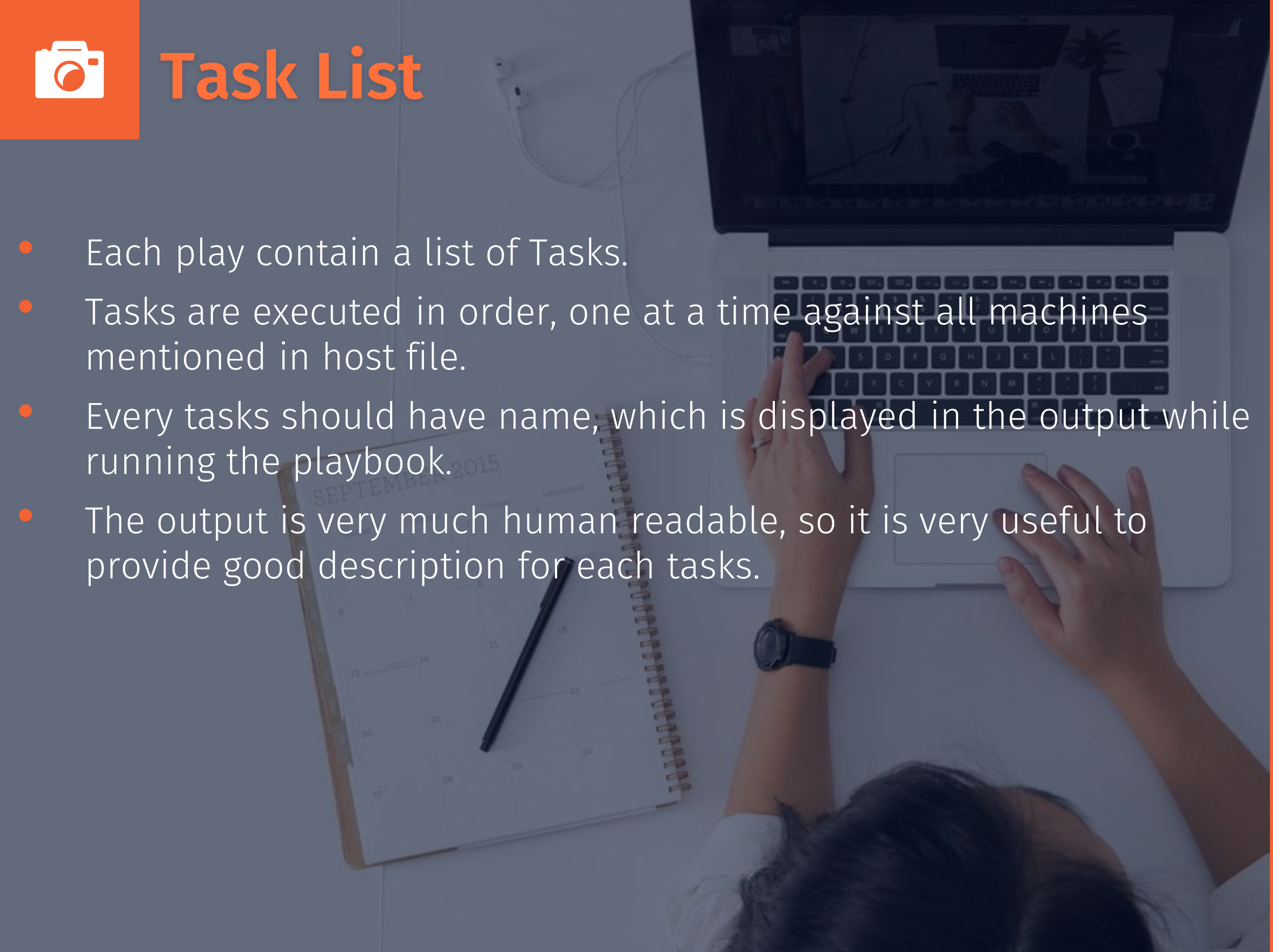
```
---
- hosts: webservers
  remote_user: yourname
  become: yes
  become_user: postgres
```

- You can also use other privilege escalation methods, like [su] if required.



```
---
- hosts: webservers
  remote_user: yourname
  become: yes
  become_method: su
```

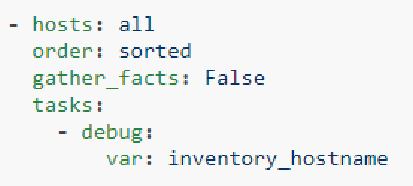
Task List



- Each play contain a list of Tasks.
- Tasks are executed in order, one at a time against all machines mentioned in host file.
- Every tasks should have name, which is displayed in the output while running the playbook.
- The output is very much human readable, so it is very useful to provide good description for each tasks.

Order control

- You can also control the order in which hosts should run.



```
- hosts: all
  order: sorted
  gather_facts: False
  tasks:
    - debug:
        var: inventory_hostname
```

- Possible values for order are:
- Inventory: The default. The order is ‘as provided’ by the inventory
- reverse_inventory: As the name implies, this reverses the order ‘as provided’ by the inventory
- sorted: Hosts are alphabetically sorted by name
- reverse_sorted: Hosts are sorted by name in reverse alphabetical order
- shuffle: Hosts are randomly ordered each run

End of this topic !

Any questions?