

OAuth2

- Back in the early days of web invention, all websites they usually ask for credentials of an end user inside an HTML form. So the end user will enter his credentials.
- The same will be sent over to the backend server inside the backend server using the credentials provided by the end user, the authentication will be completed and after the successful authentication, the backend server is going to generate a session value and the same it is going to store inside a cookie of the browser.
- So as long as the session is active, the user can access the any protected resources and URLs. So this is how the basic authentication used to work. The drawbacks of this approach is backend server will have both business logic and authentication logic tightly coupled.
- So if there is a change that you need to make inside the authentication logic, then definitely you need to make sure that it is not going to impact your business logic.
- So there has to be enough regression has to be done. Because both the business logic and the authentication logic is deployed into a single server.
- Basic authentication flow does not accommodate well the use case where users of one product or service would like to grant third-party clients access to their information on the platform.

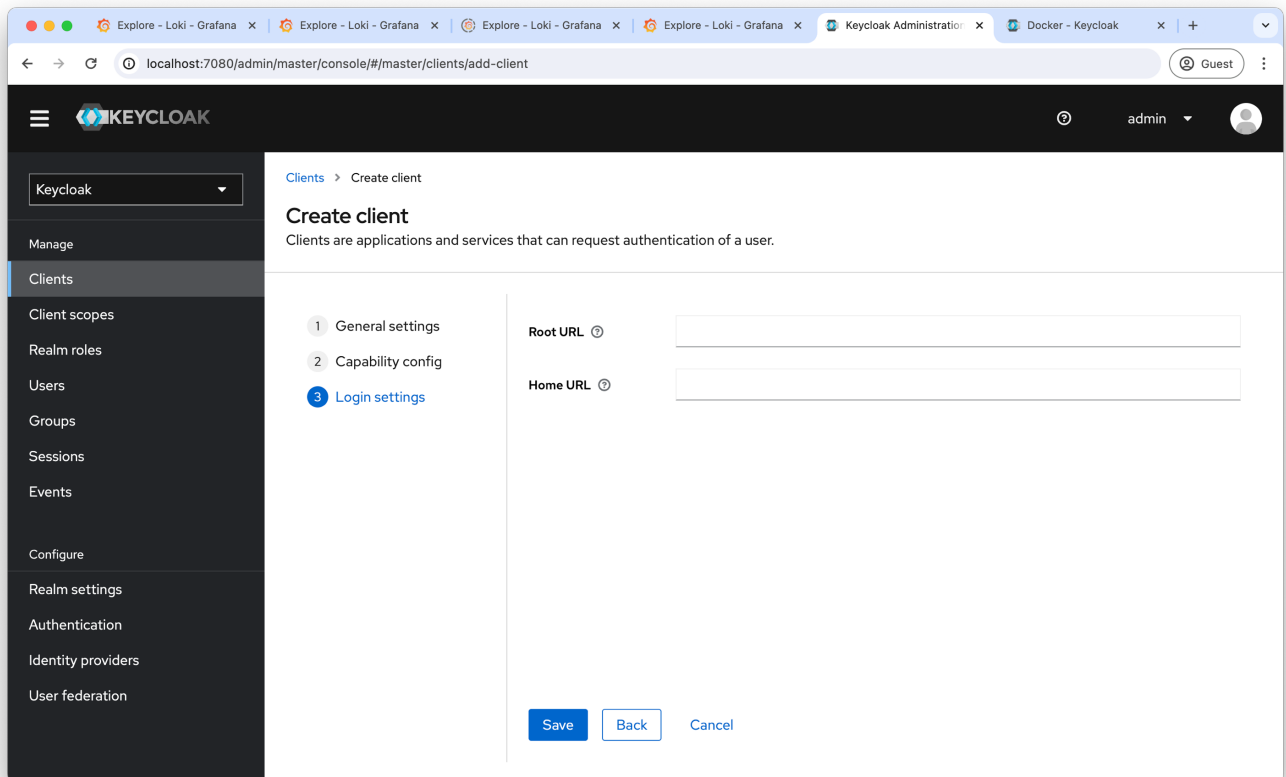
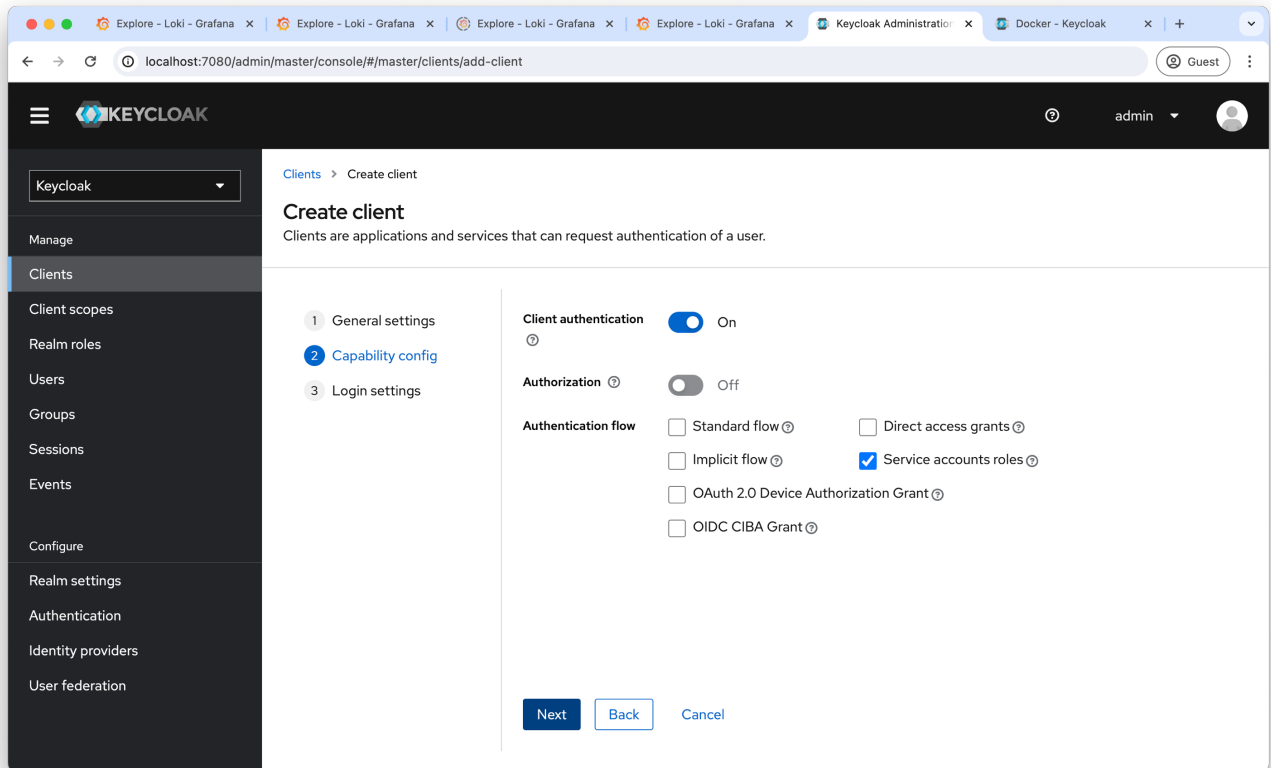
Know the difference between OpenID and OAuth2

- Review all the slides and diagrams for "Client Credentials Grant Type Flow in OAuth2"
- In Lecture 185, we are registering client details using keyClock
 - Assuming admins of keyClock and MDFinance has approved some clients coming to the application
 - Next step is to register their details using keyClock console. (Below screenshots are the sample)

The screenshot displays the Keycloak Administration console interface. The browser address bar shows the URL: `localhost:7080/admin/master/console/#/master/clients/add-client`. The page title is "Create client". Below the title, it says "Clients are applications and services that can request authentication of a user." The form is divided into two main sections: "General settings" (selected) and "Capability config". The "General settings" section includes the following fields:

- Client type**: OpenID Connect (selected from a dropdown menu)
- Client ID**: mdfinance-callcenter-cc
- Name**: MDFinance Call Center App
- Description**: MDFinance Call Center App is a service that takes the care of it's customers.
- Always display in UI**: Off (toggle switch)

At the bottom of the form, there are three buttons: "Next" (blue), "Back" (grey), and "Cancel" (blue).



Please note that the "credentials" will be generated by "keyClock". Any client that want to access the resources from MDFinance has to first get the access token from Auth Server in our case they have to authenticate through keyClock by providing client-id and client-secret. As we can see in below screenshot.

client-id: mdfinance-callcenter-cc

client-secret: S8ez9SRi6Jnam2SWo0vWhP7vm6s4yFCV

mdfinance-callcenter-cc

OpenID Connect



Enabled



Action

Clients are applications and services that can request authentication of a user.

Settings

Keys

Credentials

Roles

Client scopes

Service accounts roles

Sessions

Advanced

Client Authenticator

Client Id and Secret



Save

Client Secret

S8ez9SRi6Jnam2SWo0vWhP7vm6s4yFCV



Regenerate

Registration access
token

Regenerate

Now follow below steps to get the "token_endpoint url".

- Client has to pass this "token_endpoint url" in order to get the access token

The screenshot shows the Keycloak Admin Console interface. The left sidebar contains a menu with options: Manage, Clients, Client scopes, Realm roles, Users, Groups, Sessions, Events, Configure, Realm settings (highlighted), Authentication, Identity providers, and User federation. The main content area displays the 'Realm settings' for the 'Keycloak' realm. The settings include:

- HTML Display name:** `<div class="kc-logo-text">Keycloak</div>`
- Frontend URL:** (Empty field)
- Require SSL:** External requests
- ACR to LoA Mapping:** No ACR to LoA Mapping have been defined yet. Click the below button to add ACR to LoA Mapping, key and value are required for a key pair. [Add ACR to LoA Mapping](#)
- User-managed access:** Off
- Unmanaged Attributes:** Disabled
- Endpoints:** [OpenID Endpoint Configuration](#) and [SAML 2.0 Identity Provider Metadata](#) (both links are circled in blue with a blue arrow pointing to them)

At the bottom of the settings area, there are two buttons: **Save** and **Revert**.

We can test the flow using postman to see if we can get the access token from Auth Server

SpringBootMicroserviceWithEasybytes / Keycloak / ClientCredentials_AccessToken

POST

http://localhost:7080/realms/master/protocol/openid-connect/token

Send

Params

Authorization

Headers (8)

Body

Scripts

Tests

Settings

none

form-data

x-www-form-urlencoded

raw

binary

GraphQL

Key	Value	Description	...	Bulk Edit
<input checked="" type="checkbox"/> grant_type	client_credentials			
<input checked="" type="checkbox"/> client_id	mdfinance-callcenter-cc			
<input checked="" type="checkbox"/> client_secret	S8ez9SRi6Jnam2S0w0vWhP7vm6s4yFCV			
<input checked="" type="checkbox"/> scope	openid email profile			
Key	Value	Description		

Body

Cookies

Headers (9)

Test Results

Pretty

Raw

Preview

Visualize

JSON

```
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
```

- KeyCloak
 - POST ClientCredentials_AccessT...
- gatewayserver_security
 - POST Accounts_POST_ClientCre...
 - POST Cards_POST_ClientCreden...
 - POST Loans_POST_ClientCreden...
 - GET Accounts_GET_ClientCred...

use this request to get the token

we can use any of these request. First we need to get the access token and then use the token to hit the request. Explore the "Authorization" tab within the Postman.

Authorization Code Grant Type Flow in OAUTH2

- Understand different diagrams presented in the pdf by eazy-bytes.
- To test the demo navigate to this website: <https://oauth.com/playground/>

Create client

Clients are applications and services that can request authentication of a user.

1 General settings

2 Capability config

3 Login settings

Client type ⓘ

Client ID * ⓘ

Name ⓘ

Description ⓘ

Always display in UI ⓘ

OpenID Connect

mdfinance-callcenter-ac

MDFinance Call Center UI app

Off

Next

Back

Cancel

Create client

Clients are applications and services that can request authentication of a user.

1 General settings

2 Capability config

3 Login settings

Client authentication ⓘ

Authorization ⓘ

Authentication flow

On

Off

☒ Standard flow ⓘ

☐ Implicit flow ⓘ

☐ OAuth 2.0 Device Authorization Grant ⓘ

☐ OIDC CIBA Grant ⓘ

☐ Direct access grants ⓘ

☐ Service accounts roles ⓘ

Next

Back

Cancel

Create client

Clients are applications and services that can request authentication of a user.

1 General settings

2 Capability config

3 Login settings

Root URL ⓘ

Home URL ⓘ

Valid redirect URIs ⓘ

*

+ Add valid redirect URIs

Valid post logout redirect URIs ⓘ

+ Add valid post logout redirect URIs

Web origins ⓘ

*

+ Add web origins

Save

Back

Cancel

- Next we must create a user within KeyCloak server (set the email, username, password (temporary))
- Please note that in the latest build of the section 12 we have removed ports mapping for "accounts", "cards", "loans" services as we don't want to expose the port rather we want user/application to communicate with the gateway server/resource server in order to get access to services.
- To test the functionality, pull the latest image (12) from the docker hub and after all the containers are up and running we need to navigate to "localhost:7080" and login with admin credentials, after that we need to set up clients, realm roles, user as we can see in above snippets. Please go through postman - "gatewayserver_security" folder which has all the information about generating token (all we have to do is update the client_credentials and hit the POST request)
- For more info, please re-watch the videos.