BAB V
PENGUJIAN DAN ANALISIS

Pada bab ini dilakukan proses pengujian dan analisis terhadap Implementasi OTP Berbasis Gambar pada *Website Ecommerce* PTIIK yang telah dibangun. Proses pengujian dilakukan melalui tiga tahapan (strategi) yaitu pengujian validasi, pengujian keamanan dan pengujian *User Acceptance*. Pada pengujian validasi digunakan teknik pengujian *Black Box (Black Box Testing)*. Pada pengujian keamanan dilakukan dengan melakukan serangan *Man In The Middle* dan *Brute Force*. Pada pengujian *User Acceptance* dilakukan dengan menyebarkan kuesioner kepada user pengguna sistem.

5.1 Pengujian Validasi

Pengujian validasi digunakan untuk mengetahui apakah sistem yang dibangun sudah benar sesuai dengan yang dibutuhkan. Item - item yang telah dirumuskan dalam daftar kebutuhan dan merupakan hasil analisis kebutuhan akan menjadi acuan untuk melakukan pengujian validasi. Pengujian validasi menggunakan metode pengujian *Black Box*, karena tidak diperlukan konsentrasi terhadap alur jalannya algoritma program dan lebih ditekankan untuk menemukan konformitas antara kinerja sistem dengan daftar kebutuhan. Pada skripsi ini dilakukan pengujian validasi terhadap Impelementasi OTP Berbasis Gambar pada *Website Ecommerce* PTIIK.

5.1.1 Kasus Uji Validasi

Tabel 5.1 Kasus uji untuk pengujian validasi *login* sah untuk pembeli dan penjual

Nama Kasus Uji	Kasus Uji Login Sah
Objek Uji	SRS_002_01 dan SRS_003_01
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aplikasi dapat memenuhi kebutuhan fungsional dalam menyediakan fasilitas *login* bagi pembeli dan penjual.

Prosedur Uji	1. Pembeli dan penjual membuka *link* untuk *login.* 2. Pembeli mengisi semua *field* yang ada. Saat mengisi *field* password, pembeli diharuskan memasukkan karakter yang tertera pada gambar-gambar yang dipilih saat melakukan *register*.
Hasil yang diharapkan	Aplikasi dapat melakukan penyeleksian kondisi *login* pada *database* berdasar data yang dimasukkan dan jika penyeleksian kondisi ini benar, maka pembeli dan penjual akan mengakses ke sistem sesuai dengan hak aksesnya.

Tabel 5.2 Kasus uji untuk pengujian validasi *login* tidak sah untuk pembeli dan penjual

Nama Kasus Uji	Kasus Uji Login Tidak Sah
Objek Uji	SRS_002_01 dan SRS_003_01
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aplikasi dapat memenuhi kebutuhan fungsional dalam menyediakan fasilitas *login* bagi Penjual dan Pembeli.
Prosedur Uji	1. Pembeli dan penjual membuka *link* untuk *login.* 2. Pembeli mengisi semua *field* yang ada dan mengisikan *username* atau *password* dengan data yang salah
Hasil yang diharapkan	Aplikasi dapat melakukan penyeleksian kondisi login pada *database* berdasar data yang dimasukkan dan jika penyeleksian kondisi ini salah, maka tidak akan mengakses ke sistem dan aplikasi menampilkan pesan kesalahan.

Tabel 5.3 Kasus uji untuk pengujian validasi register

Nama Kasus Uji	Kasus Uji Register
Objek Uji	SRS_001_01
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa

	aplikasi dapat memenuhi kebutuhan fungsional dalam menyediakan fasilitas halaman untuk melakukan proses pendaftaran pembeli atau penjual
Prosedur Uji	1. Pengunjung membuka *link* untuk *register* 2. Pengunjung memilih daftar sebagai pembeli atau penjual kemudian mengisi semua *field* yang ada. Khusus untuk mengisi password, pengunjung diharuskan memilih minimal tiga gambar dan memasukkan karakter yang tertera pada gambar-gambar yang dipilih tersebut di *field password* dan konfirmasi *password*
Hasil yang diharapkan	Aplikasi menampilkan data pengunjung yang sudah terdaftar sesuai dengan *username* yang dimasukkan dan *password* gambar yang dipilih.

Tabel 5.4 Kasus uji untuk pengujian validasi mengedit akun personal

Nama Kasus Uji	Kasus Uji Mengedit Akun Personal
Objek Uji	SRS_002_02 dan SRS_003_02
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aplikasi dapat memenuhi kebutuhan fungsional dalam menyediakan fasilitas halaman untuk mengubah akun personal pembeli dan penjual.
Prosedur Uji	1. Pembeli dan penjual membuka menu *edit* akun 2. Pembeli dan penjual mengisi *field* yang ada. Saat untuk mengisi field password, pembeli dan penjual diharuskan memasukkan karakter yang tertera pada gambar-gambar yang dipilihnya
Hasil yang diharapkan	Aplikasi menampilkan halaman dengan *Username* atau *password* pembeli dan penjual ter*update* oleh *username atau password* baru yang baru saja dimasukkan.

Tabel 5.5 Kasus uji untuk pengujian validasi *reset password*

Nama Kasus Uji	Kasus Uji Mengelola *Reset Password*
Objek Uji	SRS_002_03 dan SRS_003_03
Tujuan Pengujian	Pengujian dilakukan untuk memastikan bahwa aplikasi dapat memenuhi kebutuhan fungsional dalam menyediakan fasilitas halaman untuk menyetel ulang *password* karena lupa
Prosedur Uji	1. Pembeli membuka *link* Lupa Password 2. Pembeli memasukkan alamat *email*nya yang digunakan untuk mendaftar 3. Pembeli membuka link di *email* yang telah dikirim oleh sistem 4. Pembeli mengisi field password dan konfirmasi password dengan memasukkan karakter yang tertera pada gambar-gambar yang dipilihnya
Hasil yang diharapkan	Masuk ke dalam halaman pembeli atau penjual tersebut dengan password yang baru

5.1.2 Hasil Pengujian Validasi

Tabel 5.6 Hasil pengujian validasi

No	Nama Kasus Uji	Hasil yang Diharapkan	Hasil yang Didapatkan	Status Validitas
1	Kasus Uji Login Sah	Aplikasi dapat melakukan penyeleksian kondisi *login* pada *database* berdasar data yang dimasukkan dan jika penyeleksian kondisi ini benar, maka pembeli dan penjual akan mengakses ke sistem sesuai dengan	Aplikasi dapat melakukan penyeleksian kondisi *login* pada *database* berdasar data yang dimasukkan dan jika penyeleksian kondisi ini benar, maka pembeli dan penjual akan mengakses ke sistem sesuai dengan	Valid

		hak aksesnya.	hak aksesnya.	
2	Kasus Uji Login Tidak Sah	Aplikasi dapat melakukan penyeleksian kondisi login pada *database* berdasar data yang dimasukkan dan jika penyeleksian kondisi ini salah, maka tidak akan mengakses ke sistem dan aplikasi menampilkan pesan kesalahan.	Aplikasi dapat melakukan penyeleksian kondisi login pada *database* berdasar data yang dimasukkan dan jika penyeleksian kondisi ini salah, maka tidak akan mengakses ke sistem dan aplikasi menampilkan pesan kesalahan.	Valid
4	Kasus Uji Register	Aplikasi menampilkan data pengunjung yang sudah terdaftar sesuai dengan *username* yang dimasukkan dan *password* gambar yang dipilih.	Aplikasi menampilkan data pengunjung yang sudah terdaftar sesuai dengan *username* yang dimasukkan dan *password* gambar yang dipilih.	Valid
7	Kasus Uji Edit Akun Personal	Aplikasi menampilkan halaman dengan *Username* atau *password* pembeli dan penjual ter*update* oleh *username atau password* baru yang baru saja dimasukkan.	Aplikasi menampilkan halaman dengan *Username* atau *password* pembeli dan penjual ter*update* oleh *username atau password* baru yang baru saja dimasukkan.	Valid
9	Kasus Uji Mengelola *Reset Password*	Masuk ke dalam halaman pembeli atau penjual tersebut dengan password yang baru	Masuk ke dalam halaman pembeli atau penjual tersebut dengan password yang baru	Valid

5.1.3 Analisis Hasil Pengujian Validasi

Proses analisis terhadap hasil pengujian validasi dilakukan dengan melihat konformitas antara hasil kinerja sistem dengan daftar kebutuhan. Berdasarkan hasil pengujian validasi dapat disimpulkan bahwa implementasi dan fungsionalitas sistem telah memenuhi kebutuhan yang telah dijabarkan pada tahap analisis kebutuhan.

5.2 Pengujian Keamanan

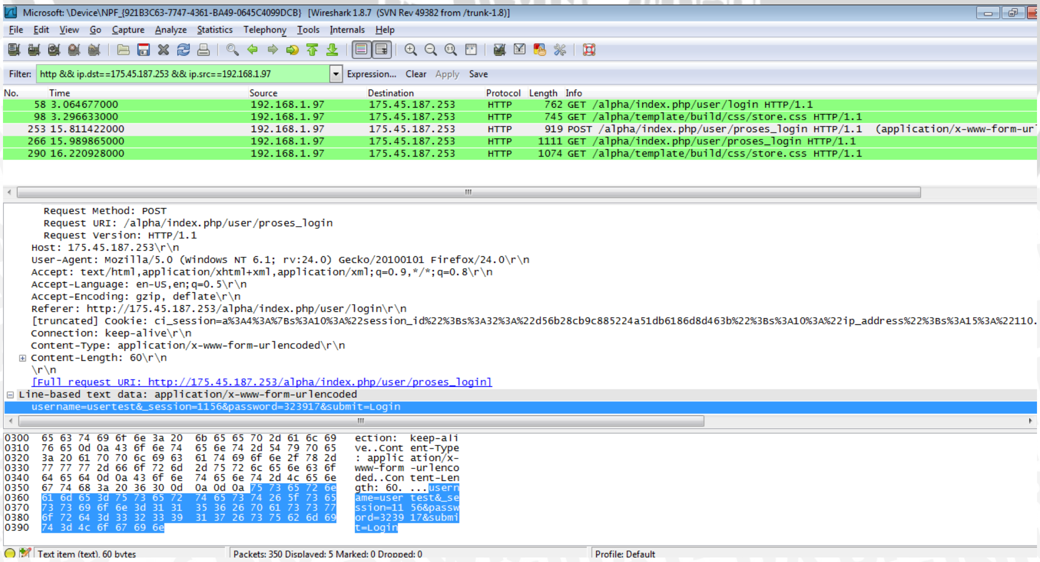
Pengujian keamanan dilakukan untuk mengetahui seberapa aman sistem yang dibangun dari serangan-serangan jaringan. Dalam penelitian ini dilakukan percobaan serangan *man in the middle* dan perhitungan kemungkinan sistem dibobol *password*nya menggunakan *brute force*.

5.2.1 Man in The Middle

5.2.1.1 Kasus Uji Man in The Middle

Pengujian dilakukan dengan menggunakan *wireshark*. *Wireshark* akan menangkap paket-paket jaringan dan menampilkan informasi paket tersebut. Pengujian dilakukan tiga kali berturut-turut saat proses *login* dengan pengguna yang sama. Berikut adalah informasi yang ditangkap oleh *wireshark* saat pengguna masuk ke dalam sistem.

Pengujian Pertama:

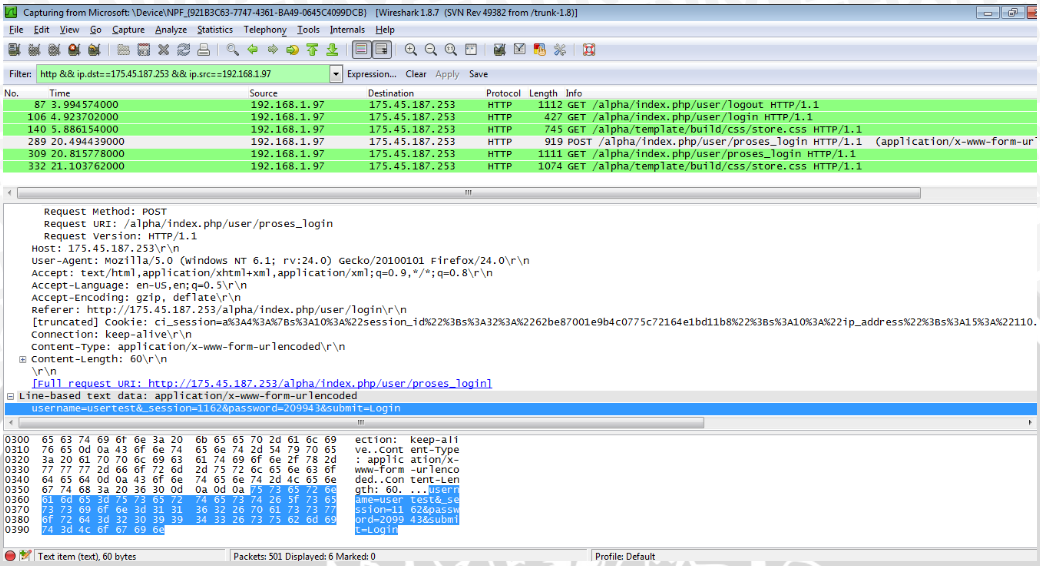


Gambar 5.1 Gambar *Printscreen* Hasil Pengujian dengan Menggunakan *Wireshark* Pengujian Pertama

Informasi yang didapat dari pengujian pertama adalah:

`username=usertest&_session=1156&password=323917&submit=Login`

Pengujian Kedua:

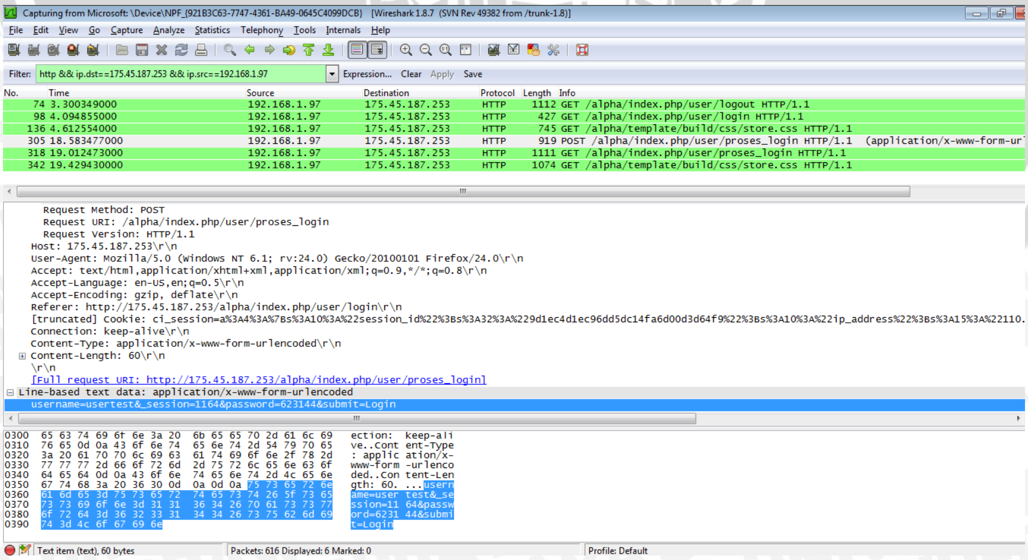


Gambar 5.2 Gambar *Printscreen* Hasil Pengujian dengan Menggunakan *Wireshark* Pengujian Kedua

Informasi yang didapat dari pengujian kedua adalah:

`username=usertest&_session=1162&password=209943&submit=Login`

Pengujian Ketiga:



Gambar 5.3 Gambar *Printscreen* Hasil Pengujian dengan Menggunakan *Wireshark* Pengujian Ketiga

Informasi yang didapat dari pengujian ketiga adalah:

```
username=usertest&_session=1164&password=623144&submit=Login
```

5.2.1.2 Analisis Hasil Pengujian Keamanan dari Serangan *Man in The Middle*

Pengujian keamanan dari serangan Man in The Middle dilakukan dengan menangkap informasi ketika pengguna melakukan proses *login*. Proses pengujian ini dilakukan selama tiga kali proses *login*. Berdasarkan hal tersebut maka dapat diambil kesimpulan bahwa dari tiga kali proses *login* yang dilakukan oleh pengguna yang sama, *password* yang diinputkan oleh pengguna berbeda-beda sehingga *password* sebenarnya tidak dapat diketahui melalui cara ini

5.2.2 Brute Force

5.2.2.1 Kasus Uji Brute Force

Pengujian dilakukan dengan dua tahap yaitu mencoba melakukan brute force terhadap sistem dengan menggunakan *software Fireforce* dan dengan menggunakan program buatan sendiri yang menggunakan CURL.

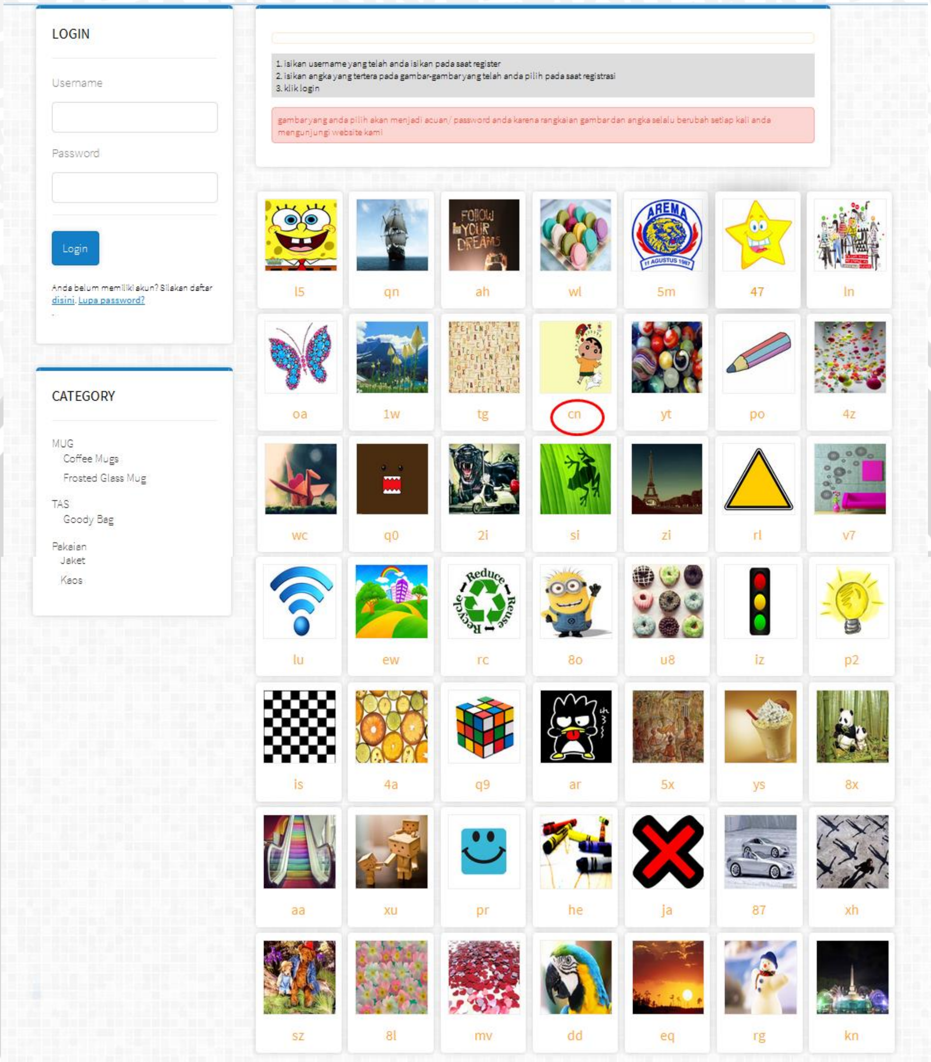
1. *Brute Force* Menggunakan *Fireforce*

Dalam percobaan pembobolan *password* ini, penulis melakukan pengujian sebanyak dua puluh kali. Penulis mengambil salah satu data pengguna yaitu pengguna yang mempunyai *username* bernama ‘aredoes’. Pengguna ‘aredoes’ ini mempunyai *password* sebagai berikut:

Tabel 5.7 Tabel Daftar Password Pengguna ‘aredoes’

Password urutan pertama	
Password urutan kedua	
Password urutan ketiga	

Urutan karakter dan gambar saat melakukan proses *brute force* adalah sebagai berikut:

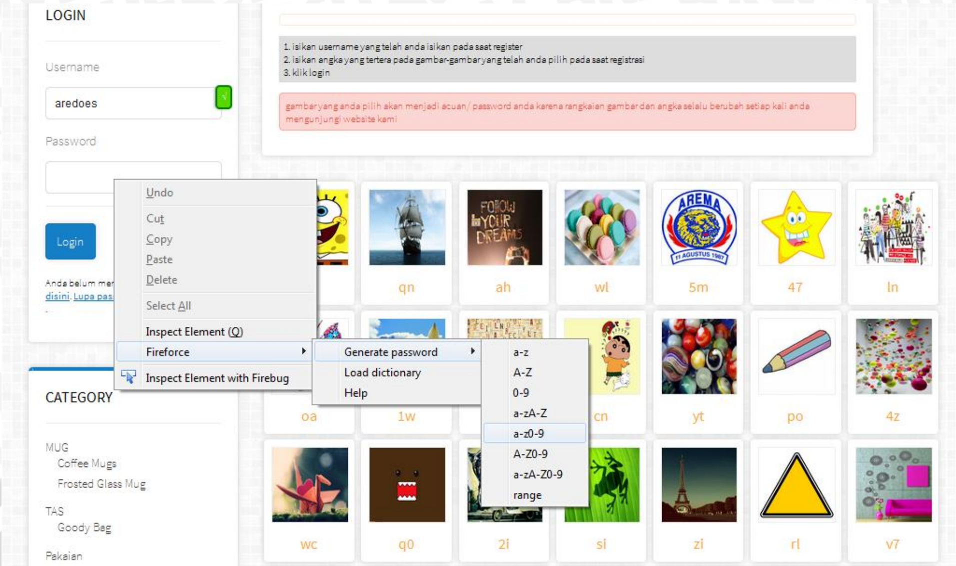


Gambar 5.4 Gambar *Printscreen* Halaman Login Saat Pengujian Brute Force

Untuk melakukan proses *brute force* dengan *Fireforce*, langkah-langkahnya adalah sebagai berikut:

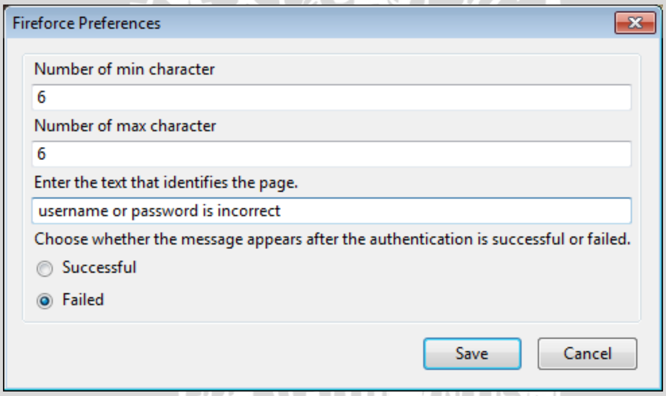
1. Memasukkan *username* pengguna
2. Klik kanan pada *field password* pilih Fireforce >> Generate Password >> a-z0-9

   Penulis memasukkan rentang karakter 0-9 dan a-z karena yang dimasukkan ke dalam *field password* adalah kombinasi antara angka dengan huruf.



Gambar 5.5 Gambar *Printscreen* Pemilihan Kombinasi Untuk Melakukan Brute Force

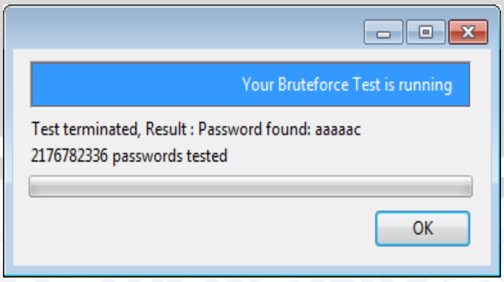
3. Penulis menguji coba panjang *password* terkecil yaitu sebanyak enam karakter atau tiga gambar



Gambar 5.6 Gambar *Printscreen* Pengisian Pengaturan Untuk Melakukan Brute Force

4. Klik save untuk melakukan proses *brute force*

Hasil yang ditemukan pada pengujian pertama adalah sebagai berikut:

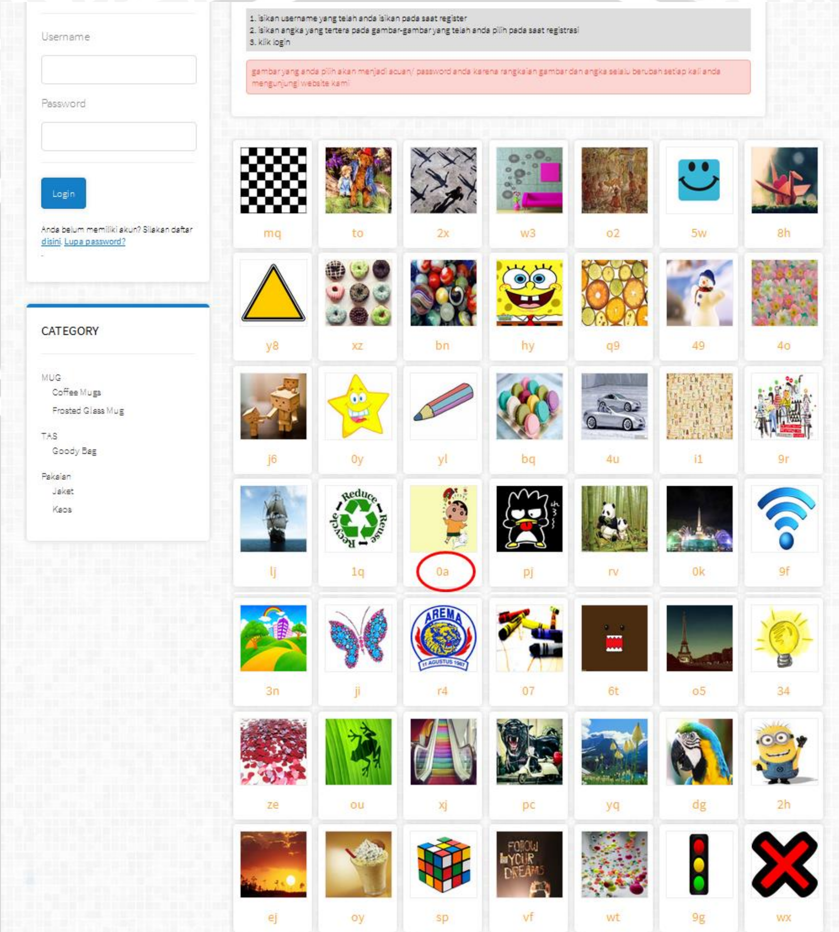


Gambar 5.7 Gambar *Printscreen* Hasil Brute Force Pengujian Pertama

Langkah-langkah di atas diulang sampai dua puluh kali pengujian.

1. Brute Force Menggunakan Program yang Dibuat dengan PHP Curl

Pengujian *brute force* dengan menggunakan program yang dibuat dengan PHP Curl ini juga menggunakan pengguna bernama ‘aredoes’ dimana saat melakukan pengujian bruteforce, susunan gambar dan karakternya adalah sebagai berikut:

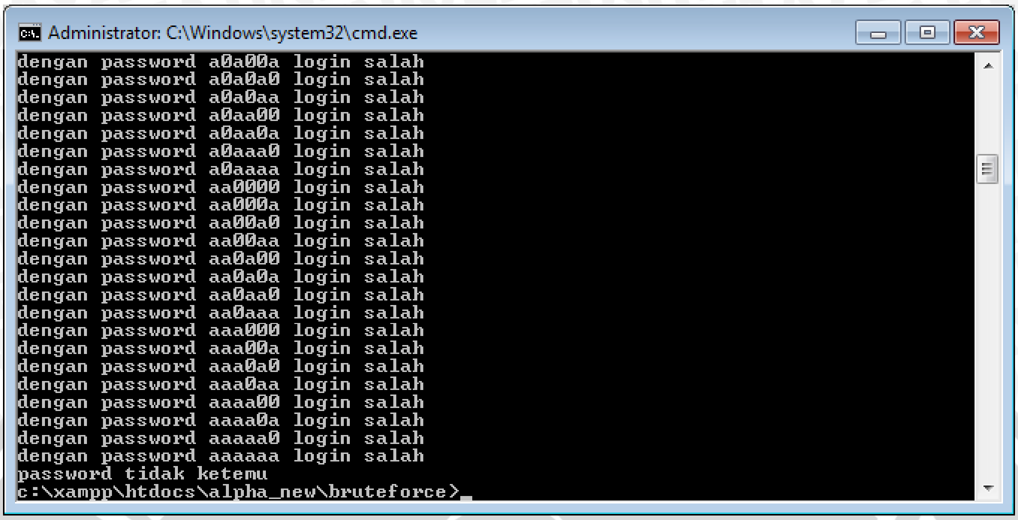


Gambar 5.8 Gambar *Printscreen* Susunan Gambar dan Karakter Ketika Melakukan Brute Force

Pengujian *brute force* menggunakan PHP Curl ini dibagi menjadi dua bagian yaitu:

a. Direct post *username* dan *password*

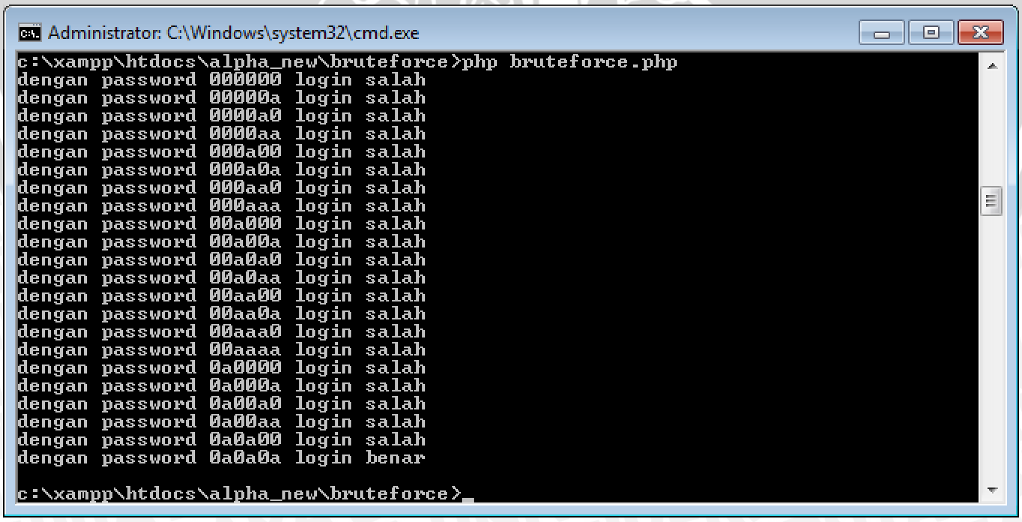
Direct post field username dan *password* ini merupakan hal yang dilakukan oleh program-program *brute force* pada umumnya, karena pada metode autentikasi web pada umumnya hanya menggunakan *username* dan *password* dalam proses autentikasinya. Hasil dari pengujian ini adalah password tidak ditemukan seperti gambar di bawah ini:



Gambar 5.9 Gambar *Printscreen* Hasil *Brute* Force dengan *Direct Post Field Username* dan *Password*

b. Direct post *username, password* dan *id session*

Direct post field username, password dan *id session* ini dilakukan karena pada metode autentikasi OTP berbasis gambar menggunakan tiga objek tersebut dalam proses autentikasinya. Hasil dari pengujian ini adalah password dapat ditemukan seperti gambar di bawah ini



Gambar 5.10 Gambar *Printscreen* Hasil *Brute* Force dengan *Direct Post Field Username, Password* dan *Id Session*

5.2.2.2 Analisis Hasil Pengujian Keamanan dari Serangan *Brute Force*

Dalam percobaan pembobolan password ini dengan menggunakan *software Fireforce*, penulis menguji sebanyak dua puluh kali dengan mencoba panjang password terkecil yaitu sebanyak enam karakter atau tiga gambar. Penguji juga mengambil salah satu data pengguna yaitu pengguna yang mempunyai *username* bernama ‘aredoes’ dengan hasil sebagai berikut.

Tabel 5.8 Tabel Daftar Hasil *Password* yang Ditemukan Melalui *Software Fireforce*

Pengujian ke-	Hasil
1	*Password found*: aaaaac
2	*Password found*: aaaaaf
3	*Password found*: aaaaab
4	*Password found*: aaaaaf
5	*Password found*: aaaaaf
6	*Password found*: aaaaaf
7	*Password found*: aaaaac
8	*Password found*: aaaaaf
9	*Password found*: aaaaab
10	*Password found*: aaaaac
11	*Password found*: aaaaab
12	*Password found*: aaaaae
13	*Password found*: aaaaae
14	*Password found*: aaaaaf
15	*Password found*: aaaaae
16	*Password found*: aaaaae
17	*Password found*: aaaaad
18	*Password found*: aaaaae
19	*Password found*: aaaaaf
20	*Password found*: aaaaaf

Dari dua puluh hasil proses pengujian, *password* yang ditemukan oleh *software fireforce* tidak berhasil membuka halaman dari pengguna ‘aredoes’. Software ini tidak dapat menemukan password sebenarnya dari pengguna ‘aredoes’ yang bernilai ‘cncncn’. Sama halnya dengan hasil pengujian dengan menggunakan *direct post username* dan *password* menunjukkan bahwa *password* tidak dapat ditemukan. Kedua pengujian ini tidak dapat menemukan *password* sebenarnya dikarenakan variabel yang diinisialisasi adalah *username* dan *password* di mana pada metode OTP berbasis gambar ini terdapat satu variabel lagi yaitu *id session*. Sehingga pada pengujian dengan *direct post username, password* dan *id session* menghasilkan bahwa *password* dapat ditemukan. Dengan ini maka *password* dari pengguna dapat ditemukan dengan catatan sebagai berikut:

a. Penyerang mengetahui metode yang digunakan oleh OTP berbasis gambar ini yaitu dengan menambahkan satu variabel *id session*
b. Tidak memuat ulang halaman web ketika melakukan proses *brute force* karena *id session* akan selalu berubah sehingga *password* yang ditemukan tidak dapat digunakan.

5.3 User Acceptance Testing

5.3.1 Kasus Uji User Acceptance Testing

UAT (*User Acceptance Testing*) digunakan untuk mengetahui apakah sistem yang dibangun telah dapat diterima atau belum oleh pengguna sistem. UAT difokuskan pada implementasi OTP yang terdapat pada proses *autentikasi*. UAT dilakukan dengan memberikan kuesioner kepada 50 mahasiswa PTIIK untuk menilai keseluruhan dan memberikan komentar maupun saran terhadap implementasi OTP tersebut. Pada skripsi ini dilakukan UAT terhadap Implementasi OTP Berbasis Gambar pada Website Ecommerce PTIIK.

Di bawah ini adalah hasil rekapitulasi jawaban dari delapan pertanyaan yang diberikan kepada responden.

Tabel 5.8 Hasil kuesioner user acceptance testing

Pertanyaan Nomor ke-	Pertanyaan	Ya	Ragu-ragu	Tidak
1	Apakah mengingat password dengan menggunakan One Time Password berbasis gambar lebih mudah diingat daripada password dengan menggunakan strong password?	37	8	5
2	Apakah meskipun anda tidak menggunakan aplikasi ini dalam jangka waktu yang cukup lama, anda akan tetap mengingat password anda?	17	26	7

3	Menurut anda, apakah sistem autentikasi menggunakan One Time Password mudah digunakan?	40	4	6
4	Menurut anda, apakah sistem autentikasi menggunakan One Time Password merupakan ide yang bagus?	39	8	3
7	Apakah anda menyukai metode One Time Password berbasis gambar daripada metode password dengan menghafal karakter-karakter seperti biasanya?	34	12	4
9	Apakah anda lebih merasa aman memiliki password dengan metode One Time Password berbasis gambar dibandingkan dengan memiliki strong password?	29	17	4
Jumlah		196	75	29

Tabel 5.9 Hasil jawaban dari pertanyaan nomor 5

Pertanyaan	Cukup	Terlalu banyak	Terlalu sedikit
Menurut anda, apakah tiga puluh enam gambar cukup untuk menjadi alternatif pilihan password?	28	11	11

Tabel 5.10 Hasil jawaban dari pertanyaan nomor 6

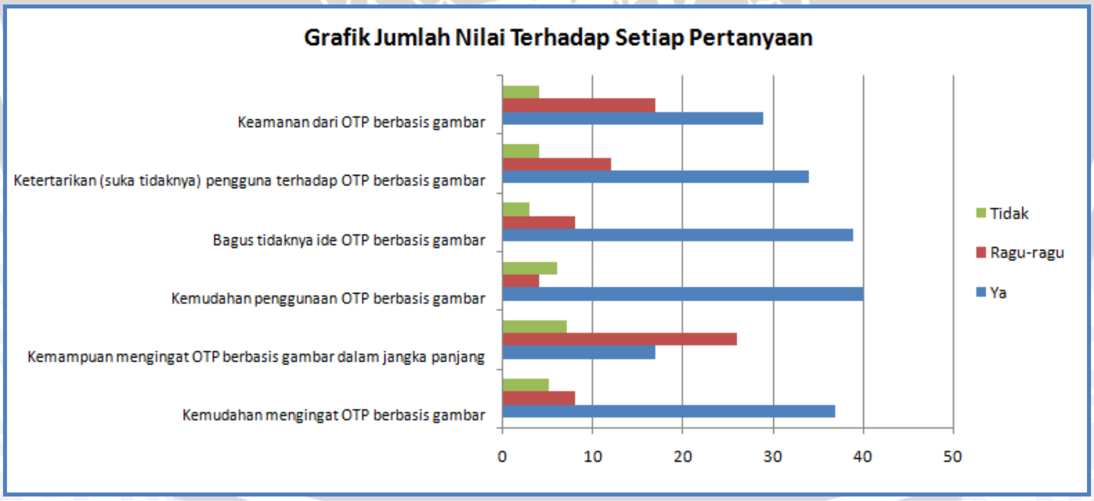
Responden	Menurut anda, berapa gambar yang bisa dijadikan alternatif pilihan password untuk pengguna?
Responden 1	100
Responden 2	20
Responden 3	9
Responden 4	100
Responden 5	150
Responden 6	16
Responden 7	50
Responden 8	13
Responden 9	30
Responden 10	100
Responden 11	36
Responden 12	24
Responden 13	9

Responden 14	10
Responden 15	20
Responden 16	9
Responden 17	10
Responden 18	50
Responden 19	9
Responden 20	10
Responden 21	25
Responden 22	160
Responden 23	12
Responden 24	9
Responden 25	99
Responden 26	30
Responden 27	40
Responden 28	64
Responden 29	50
Responden 30	9
Responden 31	9
Responden 32	128
Responden 33	36
Responden 34	25
Responden 35	25
Responden 36	25
Responden 37	25
Responden 38	100
Responden 39	100
Responden 40	24
Responden 41	16
Responden 42	20
Responden 43	30
Responden 44	100
Responden 45	9
Responden 46	20
Responden 47	30
Responden 48	40
Responden 49	50
Responden 50	50
Rata- rata	**42,7**

5.3.2 Analisis Hasil User Acceptance Testing

Proses analisis terhadap hasil pengujian sistem terhadap pengguna dilakukan dengan menghitung jumlah tiap nilai dari semua koresponden. Hasil penilaian terhadap pengguna mendapatkan nilai yang ditunjukan pada Gambar 5.7.

Pada Gambar 5.7 diperoleh bahwa 74% responden berpendapat bahwa password dengan metode OTP berbasis gambar ini mudah diingat daripada *strong password*, namun sebanyak 52% responden ragu-ragu dapat mengingat password dengan metode OTP berbasis gambar ini dalam jangka waktu yang panjang. Hal ini disebabkan gambar kurang umum sehingga susah diingat. Sebuah penelitian menunjukkan bahwa visual memory sesorang dapat bersifat subjektif namun ada sesuatu yang membuat sebuah gambar mudah diingat [SDW-11]. Kemudian dalam tingkat kemudahannya, sebanyak 80% responden menyatakan bahwa OTP berbasis gambar mudah digunakan. Kemudian sebanyak 78% responden berpendapat bahwa OTP berbasis gambar ini merupakan ide yang bagus. Responden menyukai metode OTP berbasis gambar sebanyak 68%. Dan responden yang merasa aman dengan metode OTP berbasis gambar sebanyak 58%

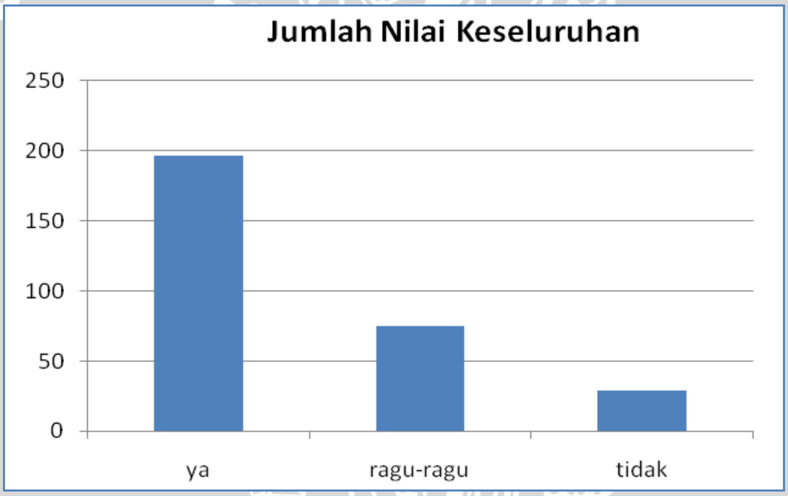


Gambar 5.8 Grafik jumlah nilai terhadap setiap pertanyaan

Selanjutnya terkait dengan jumlah pilihan gambar (pertanyaan nomor 5), sebesar 56% responden menyatakan bahwa 36 gambar cukup untuk menjadi alternatif pilihan *password*, dan sebesar 22% responden menyatakan bahwa 36 gambar terlalu banyak untuk menjadi alternatif pilihan *password*, sebesar 22% responden juga menyatakan bahwa 36 gambar terlalu sedikit.

Untuk pertanyaan "Menurut anda, berapa gambar yang bisa dijadikan alternatif pilihan *password* untuk pengguna?", dari 50 jawaban yang diberikan responden, diambil rata-rata yaitu sebesar 43 pilihan gambar. Namun dikondisikan dengan tata letak gambar pada halaman *website*, maka peneliti memutuskan untuk membulatkan ke bilangan kuadrat terdekat yaitu sebesar 49 gambar agar dapat di posisikan tujuh gambar mendatar dan tujuh gambar menurun.

Pada Gambar 5.8 merupakan jumlah keseluruhan dari tiap penilaian. Hasil dari penjumlahan nilai keseluruhan dapat diambil kesimpulan bahwa sebesar 65,3% responden dapat menerima sistem ini, sebesar 34,7% responden kurang dapat menerima sistem ini.



Gambar 5.9 Grafik jumlah keseluruhan dari tiap nilai

Secara umum responden merespon dengan baik metode OTP berbasis gambar ini. Para responden yang menyukai metode OTP berbasis gambar ini berpendapat bahwa metode ini merupakan inovasi yang menarik dan sesuatu yang visual lebih mudah diingat dibandingkan dengan sesuatu yang bersifat teks atau numerik. Meskipun demikian perlu diperhatikan bahwa dalam pengimplementasian OTP berbasis gambar ini masih ada pengguna yang belum dapat menerima karena beberapa alasan, diantaranya *password* tiap gambar saat *login* selalu berubah sehingga *user* perlu melihat satu persatu, gambarnya kurang umum sehingga susah untuk diingat dan masih belum terbiasa