

Seit Veröffentlichung meines Artikels zur SAP-Passwort-Sicherheit <http://www.it-audit.de/assets/artikel/eigen/SAP-Passwort.pdf> ist ein gutes Jahr vergangen – Zeit für ein Review.

Anmerkungen zum im August 2003 veröffentlichten Artikel

Schutz des im Klartext eingegebenen Passworts

Ich habe in meinem Artikel nicht alle mir bekannten Möglichkeiten erwähnt, wie man an die Klartext-Passworte gelangt. Auch ohne den Quelltext des Programms SAPMSYST zu ändern, lässt sich das bei der Anmeldung eingegebene Passwort abfangen. Das Aufzählen der anderen mir bekannten Möglichkeiten würde jedoch einer unmittelbaren, von jedem Anfänger nachvollziehbaren Anleitung zum Abfangen der SAP-Passworte gleichkommen. Deswegen habe ich darauf verzichtet, diese weiteren Möglichkeiten im Artikel zu nennen.

Eine Änderung des Passwortes während der Anmeldung ist jedoch noch sicherer als die seit Release 6.20 mögliche Änderung des Passwortes in Transaktion SU3 (System -> Benutzer -> Eigene Daten).

Knacken von SAP-Passworten per Dictionary- oder Brute-Force-Attacke

Die Angaben zur Geschwindigkeit, mit der sich SAP-Passworte knacken lassen, sind inzwischen veraltet.

Per ABAP-Programm konnte ich noch eine Steigerung von 7,5 Millionen Kombinationen auf 8 Millionen Kombinationen pro Minute erreichen.

Mit Hilfe eines Patches für das Passwort-Crack-Programm John the Ripper¹ kann ich über 90 Millionen Kombinationen pro Minute ausprobieren - auf dem gleichen System (Athlon XP2100+), das auch für die Geschwindigkeitsmessung mit einem ABAP-Programm genutzt wurde.

Das heißt, SAP-Passworte lassen sich ca. 300 mal so schnell wie FreeBSD-Passworte und 5000 mal so schnell wie OpenBSD-Passworte knacken.

Die Anzahl der geknackten Passworte wird sich dadurch allerdings nur unwesentlich erhöhen, da die meisten Passworte so trivial sein dürften, dass sie sich auch mit einem ABAP-Programm in kurzer Zeit knacken lassen. (In Bruce Schneiers Buch "Secrets & Lies. Digital Security in a Networked World" wird eine Studie erwähnt, die ermittelte, dass 86 Prozent der Passworte leicht zu knacken waren.)

Der Hauptvorteil bei der Nutzung von John the Ripper besteht also eher darin, dass man auf die bereits implementierten ausgefeilten Strategien zur Ermittlung der aussichtsreichsten Passwort-Kandidaten (diverse "password mangling rules" für Dictionary-Attacken und "incremental modes" basierend auf statistischen Informationen zur Häufigkeit von Zeichen und Zeichenkombinationen in Passwörtern) zurückgreifen kann.

¹ <http://www.openwall.com/john/>

Änderungen im SAP-Standard

Prüfung der Passwort-Historie während der Passwort-Änderung

Mit OSS-Hinweis 732038 (und den darin genannten SAP-Kernel-Patches) wollte SAP das Problem beheben, dass bei einer Passwort-Änderung durch den Benutzer der älteste in USR02 gespeicherte Hashwert (USR02-OCOD5) nicht überprüft wurde. (Anmerkung: OSS-Hinweis 2467 erwähnt, dass der Benutzer bei einer Passwort-Änderung die letzten fünf Passworte nicht verwenden darf.

Das kann man auch so verstehen, dass das aktuelle Passwort und die vier vorhergehenden Passworte nicht verwendet werden dürfen.

Dann hätte das bisherige Verhalten der Beschreibung in OSS-Hinweis 2467 entsprochen.)

Mit dem Kernel-Patch werden jetzt zwar der älteste Hashwert (USR02-OCOD5) und die Hashwerte USR02-OCOD1 - USR02-OCOD4 überprüft, jedoch wurde dabei ein neuer Fehler eingebaut: Der zum bisher aktuellen Passwort gehörende (und in USR02-BCODE gespeicherte) Hashwert wird nicht mehr überprüft.

Das heißt, ein Benutzer kann bei der Passwort-Änderung als neues Passwort das aktuelle Passwort wiederverwenden. Erst bei der darauffolgenden Passwort-Änderung muss er sich ein neues Passwort ausdenken, da der zuvor nur in USR02-BCODE stehende Hashwert seit der Wiederverwendung des alten Passwortes auch in USR02-OCOD1 steht. Man kann also jedes Passwort zwei mal verwenden (und damit jedes Passwort doppelt so lange nutzen wie im Profil-Parameter login/password_expiration_time festgelegt.) Auch die Mindestanzahl unterschiedlicher Passworte, die ein Benutzer sich merken muss, verringert sich dadurch.

In Releases ≥ 6.10 wird dieser Fehler durch den in älteren Releases nicht vorhandenen Profil-Parameter login/min_password_diff in seinen Auswirkungen gemildert. Der Default-Wert für login/min_password_diff ist 1, das neue Passwort muss sich also in mindestens einem Zeichen vom alten Passwort unterscheiden. Jedoch ist es auch hier möglich, ein neues Passwort zu wählen, das den gleichen Hashwert hat wie das bisher aktuelle Passwort. Der Nutzer kann also als neues Passwort SÖHNEXYZ wählen, wenn sein bisheriges Passwort SÜHNEXYZ war, obwohl beide Passworte den gleichen Hashwert haben.

Fehlende Unterscheidung von ^ und nicht zum 7bit-ASCII-Zeichensatz gehörenden Zeichen

Mit OSS-Hinweis 735356 (und den darin genannten SAP-Kernel-Patches) hat SAP auf das Problem von Kollisionen im Hash-Algorithmus reagiert, die auftraten, wenn das Zeichen '^' (Accent Circumflex) oder nicht zum 7bit-ASCII-Zeichensatz gehörende Zeichen im Passwort verwendet wurden.

Dieser Hinweis erwähnt, welche Zeichen im Passwort verwendet werden können, ohne dass bei CODVN B (für einen Angreifer ausnutzbare) Hash-Kollisionen auftreten. Außerdem wurde ein neuer Profil-Parameter login/password_charset eingeführt.

Der Default-Wert login/password_charset = 1 sorgt dafür, dass sich das System genauso verhält wie mit älteren Kernel-Patches. Als Hash-Algorithmus wird CODVN B verwendet, die Verwendung von '^' oder von nicht zum 7bit-ASCII-Zeichensatz gehörenden Zeichen führt zu Hash-Kollisionen.

Der Wert login/password_charset = 0 sorgt dafür, dass '^' sowie nicht zum 7bit-ASCII-Zeichensatz gehörende Zeichen nicht mehr in neu vergebenen Passwörtern verwendet werden dürfen, als Hash-Algorithmus wird weiterhin CODVN B verwendet. Damit werden die für einen Angreifer vorhersagbaren Hash-Kollisionen in Passwörtern vermieden.

Um die durch fehlende Unterscheidung von '^' und nicht zum 7bit-ASCII-Zeichensatz gehörenden Zeichen verursachten Hash-Kollisionen zu vermeiden, hätte es allerdings ausgereicht, nur die nicht zum 7bit-ASCII-Zeichensatz gehörenden Zeichen zu verbieten.

Aber weil auch '^' nicht mehr zugelassen ist, muss ein Angreifer nur noch 68 verschiedene Zeichen (statt bisher 69) berücksichtigen.

Die Gesamtzahl der möglichen Passwörter verringert sich dadurch um ca. 11%. Jedoch würde es nach wie vor mehrere Jahre dauern, bis ein Angreifer (mit einem PC) alle möglichen Passwort-Kombinationen für einen Usernamen ausprobiert hat.

Dadurch, dass weiterhin CODVN B als Hash-Algorithmus verwendet wird, bleibt mit login/password_charset = 0 das Problem bestehen, dass z.B. für die zwei Benutzer P_MÜLLER und P_MÖLLER bei Verwendung des gleichen Passwortes der gleiche Hashwert erzeugt wird, was sich von einem Angreifer beim Knacken von SAP-Passwörtern ausnutzen lässt, da die Berechnung des Hashwertes pro Passwort nur einmal ausgeführt werden muss.

Versucht ein Anwender, ein bei login/password_charset = 0 nicht mehr zugelassenes Zeichen im neu vergebenen Passwort zu verwenden, wird leider eine wenig hilfreiche Fehlermeldung angezeigt.

(“Bitte nur Zeichen aus dem 'syntactical Characterset' verwenden“ – Message-Nummer 00 185)

Wenigstens im Langtext der Fehlermeldung sollten die erlaubten Zeichen explizit einzeln aufgeführt werden.

Stattdessen ist auch hier nur erwähnt, dass neben Ziffern 0-9 und den Buchstaben des US-Alphabets nur noch "verschiedene Sonderzeichen, die auf allen Tastaturen zu finden sind", erlaubt sind.

Ein Anwender weiß jedoch üblicherweise nicht, welche Zeichen nicht zum 7bit-ASCII-Zeichensatz gehören. Zulässige Zeichen wie {, [,], }, ~, |, @ dürften für ihn eher als auf einer Tastatur schwerer zu erreichen gelten als z.B. deutsche Umlaute oder das §-Zeichen.)

Bei login/password_charset = 2 sind auch '^' und nicht zum 7bit-ASCII-Zeichensatz gehörende Zeichen in Passwörtern erlaubt, es wird ein neuer Hash-Algorithmus (CODVN D) verwendet.

Damit dieser neue Hash-Algorithmus verwendet werden kann, sind unbedingt auch die in OSS-Hinweis 735356 erwähnten ABAP-Sourcecode-Korrekturen einzubauen! (Der Report RSUSR003, der nach SAP-Standard-Benutzern mit allgemein bekannten Default-Passwörtern sucht, wurde jedoch nicht per Vorab-Korrektur angepasst. Wenn also nach Änderung des Profil-Parameters login/password_charset auf 2 für die User DDIC, EARLYWATCH, SAP* oder SAPCPIC die Passwörter auf die jeweiligen Default-Passwörter geändert werden, wird dies nicht erkannt, weil das Programm nur die Hashwerte für CODVN A und CODVN B überprüft.)

Beim neuen Hash-Algorithmus (CODVN D) werden die für einen Angreifer leicht vorhersagbaren Hash-Kollisionen bei Verwendung der o.g. Zeichen vermieden. (Obwohl es im OSS-Hinweis 735356 nicht explizit erwähnt ist, werden auch die Hash-Kollisionen, die durch nicht zum 7bit-ASCII-Zeichensatz gehörende Zeichen im Benutzernamen verursacht wurden, mit CODVN D vermieden. Bei gleichem Passwort wird also mit CODVN D für die Benutzer P_MÜLLER und P_MÖLLER ein anderer Hashwert ermittelt.)

Dazu wird vor der Berechnung des Hashwertes eine UTF-8-Konvertierung des Benutzernamens und des Passwortes vorgenommen. Obwohl die fehlende Unterscheidung von Groß- und Kleinbuchstaben weiterhin bestehen bleibt und auch für nicht zum 7bit-ASCII-Zeichensatz gehörende Zeichen gilt, erhöht sich damit die Anzahl der in einem Passwort verwendbaren Zeichen. Statt bisher 69 Zeichen muss ein Angreifer bis zu 66 weitere Zeichen (0xA0-0xDF, 0xF7, 0xFF - s.

<http://www.unicode.org/charts/PDF/U0080.pdf>) berücksichtigen, womit sich fast 218 mal so viele mögliche Passwörter wie mit CODVN B ergeben, vorausgesetzt ein Anwender kann alle entsprechenden Zeichen an den von ihm genutzten Frontends eingeben.

(Hinweis: Zeichen, die in dem auf dem Frontend-PC verwendeten Zeichensatz anders codiert sind als in der vom Applikationsserver verwendeten Code-Page, können nicht verwendet werden. Dies betrifft z.B. das €-Zeichen)

Leider hat SAP weder den Algorithmus zur Ermittlung des Hashwertes so verändert, dass sich die zur Berechnung eines Hashwertes nötige Zeit erhöht, noch wurde eine Unterscheidung von Groß- und Kleinschreibung eingeführt.

Neue Erkenntnisse zu Fehlern im Hash-Algorithmus

Sporadisch werden für Passworte, die in den ersten 7 Bytes identisch sind, gleiche Hashwerte berechnet. Wie oft derartige Kollisionen auftreten, hängt von der Länge des Benutzernamens ab.

Da der neue Hash-Algorithmus CODVN D weitestgehend mit dem Hash-Algorithmus CODVN B identisch ist, ist auch der neue Hash-Algorithmus betroffen. (Durch die UTF-8-Konvertierung sind die Auswirkungen für CODVN D sogar gravierender als für CODVN B.)

Für Namen mit einer Länge von weniger als 3 Zeichen habe ich mit CODVN B keine solchen Kollisionen gefunden. Bis zu einer Länge von 7 Zeichen nimmt die Häufigkeit von Kollisionen zu. Für längere Namen steigt die Häufigkeit nicht weiter an (solange nicht CODVN D verwendet wird und im Passwort Zeichen > 0x7F vorkommen).

Um einen überschaubaren, aber signifikanten Anteil der Passworte auf entsprechende Kollisionen zu überprüfen, habe ich für verschiedene Benutzernamen die Hashwerte zu folgenden Passworten ermittelt: "AABAA01 " - "ZZYZZ12~".

Die ersten 5 Zeichen des Passwortes enthalten die Buchstaben A-Z (ohne Kombinationen, in denen die 3 Anfangsbuchstaben identisch sind).

Die Stellen 6 und 7 sind jeweils 01, 02, 03, 04, 05, 06, 07, 08, 09, 10, 11 und 12.

Das letzte Zeichen ist ein beliebiges der 69 verschiedenen für CODVN B relevanten Zeichen (also mit '^' und ' ', aber ohne Zeichen, die nicht zum 7bit-ASCII-Zeichensatz gehören).

Pro Benutzer wurden also $(26 \cdot 26 \cdot 26 - 26) \cdot 26 \cdot 26 \cdot 12 \cdot 69 = 818.602.200$ Passworte überprüft.

Es wurden nicht alle gefundenen Hashwerte gespeichert und miteinander verglichen, sondern nur jeweils die Hashwerte eines Benutzers für die Passworte, die in den ersten 7 Bytes übereinstimmen.

Das Ergebnis ist in den folgenden Tabellen dargestellt, jeweils für CODVN B und für CODVN D.

Die erste Spalte enthält die SAP-Benutzernamen, in der zweiten Spalte ist die Anzahl der Passworte aufgeführt, für die es mindestens ein weiteres Passwort mit identischem Hashwert gibt.

Die gefundenen Hashwert-Kollisionen sind in der überwiegenden Mehrzahl Kollisionen zwischen genau zwei Passworten. Nur vereinzelt gab es Kollisionen zwischen mehr als zwei Passworten. Dies ist durch einen Vergleich mit der dritten Spalte zu erkennen, die die Anzahl der Hashwerte enthält, für die Kollisionen gefunden wurden.

Die vierte Spalte enthält die Anzahl der 7stelligen Passworte, für die mindestens ein in den ersten 7 Zeichen identisches 8stelliges Passwort gefunden wurde, das den gleichen Hashwert hat.

Übersicht Hash-Kollisionen für CODVN B:

Benutzername	Anzahl Passworte mit Kollisionen	Anzahl Hashwerte	7stellige Passworte
123	16	8	0
ABC	4	2	0
SAP	4	2	0
DDIC	332	166	3
SAP*	344	172	0
12345	5218	2609	20
ABCDE	5315	2656	17
ABCDEF	49369	24659	264
TMSADM	49007	24481	228
1234567	319388	159439	2090
ABCDEFG	320140	159818	2112
DITTRICH	320807	160143	2127
DEVELOPER	320918	160220	2090
ABCDEFGH	321771	160618	2114
ABCDEFGHIJ	321332	160402	2250
EARLYWATCH	321872	160691	2256
12345678901	320088	159791	2101
ABCDEFGHIJK	320637	160062	2116
ABCDEFGHIJKL	319449	159482	2127
PASSWORDTEST	319711	159613	2074

Das heißt für Benutzernamen mit mindestens 7 Zeichen Länge, dass es ca. zu jedem 2550. Passwort ein weiteres in den ersten 7 Zeichen übereinstimmendes Passwort mit identischem Hashwert gibt (ohne die Hashwert-Kollisionen durch nicht zum 7bit-ASCII-Zeichensatz gehörende Zeichen).

Und bei ca. jedem 380000. 8stelligen Passwort kann man das letzte Zeichen weglassen und erhält dennoch den gleichen Hashwert.

Da ein Benutzer nicht weiß, für welche Passworte solche Kollisionen auftreten, muss er sicherheitshalber solche 8-stelligen Passworte wählen, die selbst nach Ersetzen des 8. Zeichens durch ein beliebiges anderes Zeichen oder nach Kürzung auf 7 Zeichen noch schwer zu knacken sind. Denn selbst wenn per Profile-Parameter die minimale Passwortlänge auf 8 Zeichen festgelegt ist, kann dies nur bei der Vergabe eines neuen Passwortes, aber nicht beim Login geprüft werden. Also ist auch mit dem 7stelligen Passwort eine Anmeldung möglich, wenn der Hashwert identisch zu dem vom Benutzer vergebenen 8stelligen Passwort ist. Um alle maximal 7stelligen Passworte auszuprobieren, brauche ich pro Benutzer auf meinem PC z.B. weniger als 2 Monate.

Durch diese Hash-Kollisionen kann also selbst ein absolut zufälliges 8stelliges Passwort geknackt werden, auch wenn der Angreifer nur 7stellige Passworte ausprobiert.

Beispiele von Hash-Kollisionen für CODVN B

Benutzername	Hashwert	Passwort	
(Kollisionen mit 7stelligen Passwörtern, Kollisionen von mit 'A' beginnenden Passwörtern, Kollisionen von mit 'Z' beginnenden Passwörtern)	74DB83791A028420	DFQEX12 DFQEX12.	
	DB85CC4AC17CE1CC	FJZRE06 FJZRE06.	
	922C8AD944ED9C87	YJIKU07 YJIKU07P	
	4E13F3A73D9EEBDB	AJPRE01* AJPRE01V	
	44E00197CC9A06F7	AZVUM12- AZVUM12D	
	2A9FF2C0B853FADF	ZCRBL054 ZCRBL05R	
	5CB085C4E30C5BD7	ZGWIY129 ZGWIY12:	
	92383AA2964288F3	ZHIBI115 ZHIBI11}	
	3F8604378562E643	ZPTJW01B ZPTJW01[
	BF8DCBC87E5AA0C8	ZSFAR08% ZSFAR08:	
	627635A34CB882D0	ZXHIH01Q ZXHIH01X	
	729C96D5E9093794	ZXULK099 ZXULK09:	
	SAP* (Kollisionen von mit 'A' beginnenden Passwörtern)	FE9A3CDFB52333FF	ACDBE02& ACDBE02
		22AEBD2708D11FEB	ADCPD08] ADCPD08
		26ECF1B0B4EFB601	AETOX04 , AETOX04E
		B0353ABEC53AEC0C	APOES01 (APOES014
		AB2CA4709CB7FA7F	ASVNE03D ASVNE03 }
		BD1510AD59D88210	AUTKA033 AUTKA035
8DA7085B3B04DBD8		AVEGC12 (AVEGC12 <	
E5DF8AB7E279CC9E		AYNRS120 AYNRS12H	

EARLYWATCH (Kollisionen von jeweils 4 verschiedenen Passwörtern, Kollisionen von mindestens 2 verschiedenen 8stelligen Passwörtern mit einem 7stelligen Passwort)	C1490E1C2AC53FFB	COCQP098
		COCQP09E
		COCQP09J
		COCQP09V
	E786D382B2C88932	VXFNI07+
		VXFNI07<
		VXFNI07V
		VXFNI07W
	7A009B2ED709FD77	BKEFR06
		BKEFR06I
		BKEFR06X
	5A372809E0119B7F	HDRIE10
		HDRIE10)
		HDRIE10G
	FB031631D255FBED	SBVLK09
		SBVLK09+
		SBVLK09S
	2F663DD1010BD94E	SUNMS07
		SUNMS07!
		SUNMS07G
	8C68CDD9BE184AE1	THZZN08
		THZZN08 (
		THZZN08H
	5BCDD8FB7B827A26	VAUBS04
VAUBS04*		
VAUBS04H		

Es ist davon auszugehen, dass es auch für zwei verschiedene Usernamen, die in den ersten 7 Zeichen übereinstimmen, Passwörter gibt, für die die jeweiligen Hashwerte identisch sind.

Da diese Kollisionen aber nur sporadisch auftreten, dürften sie kaum von einem Angreifer ausgenutzt werden können.

Daher habe ich keine diesbezüglichen Tests vorgenommen.

Übersicht Hash-Kollisionen für CODVN D:

Benutzername	Anzahl Passworte mit Kollisionen	Anzahl Hashwerte	7stellige Passworte
123	10	5	0
ABC	12	6	0
SAP	8	4	0
DDIC	316	158	0
SAP*	350	175	1
12345	5309	2654	21
ABCDE	5362	2680	15
ABCDEF	49287	24615	236
TMSADM	49165	24563	253
1234567	321170	160339	2066
ABCDEFG	320840	160162	2182
DITTRICH	320945	160143	2138
DEVELOPER	321014	160215	2190
ABCDEFGH	321072	160248	2128
ABCDEFGHI	320916	160307	2138
EARLYWATCH	321091	160287	2151
12345678901	321879	160467	2140
ABCDEFGHIJK	321452	160214	2125
ABCDEFGHIJKL	321265	160380	2122
PASSWORDTEST	320997	160691	2107

Für CODVN D sind die Ergebnisse mit denen aus CODVN B vergleichbar. Lediglich bei sehr kurzen Benutzernamen gibt es geringfügige Abweichungen, die sich aber reduzieren dürften, wenn man eine größere Anzahl Passworte bzw. weitere 3stellige Benutzernamen untersucht.

Es ist allerdings zu beachten, dass hier für CODVN D nur die bei CODVN B verwendeten Zeichen überprüft wurden.

Wenn man an der 8. Stelle auch nicht zum 7bit-ASCII-Zeichensatz gehörende Zeichen verwendet, nimmt die Anzahl von Kollisionen noch deutlich zu, obwohl die Anzahl der Passworte sich dann nicht einmal verdoppelt.

Dies wird anhand der folgenden Tabelle deutlich.

Wegen der größeren Anzahl von Mehrfach-Kollisionen (drei oder mehr Passworte mit identischem Hashwert) wurde hier eine etwas detailliertere Darstellung gewählt.

Übersicht Hash-Kollisionen für CODVN D
(als 8. Zeichen 66 weitere zulässige Zeichen verwendet):

Benutzername	Anzahl PW/ Hashwert	Anzahl Hashwerte	7stellige Passworte	Anzahl Passworte
123	2	3069	0	6138
				6138
ABC	2	3055	0	6110
				6110
SAP	2	3083	0	6166
				6166
DDIC	2	31976	0	63952
	3	13	0	39
				63991
SAP*	2	32175	1	64350
	3	11	0	33
				64383
12345	2	209868	47	419736
	3	192	0	576
				420312
ABCDE	2	210139	29	420278
	3	222	0	666
				420944
ABCDEF	2	950165	492	1900330
	3	1804	0	5412
	4	6	0	24
				1905766
TMSADM	2	949154	505	1898308
	3	1789	3	5367
	4	2	0	8
				1903683
1234567	2	3300995	4035	6601990
	3	10722	45	32166
	4	34	0	136
				6634292
ABCDEFGF	2	3299952	4194	6599904
	3	10794	46	32382
	4	39	0	156
				6632442
DITTRICH	2	5363716	4181	10728246
	3	17715	39	53145
	4	57	0	228
				10780793
ABCDEFGFHI	2	5366661	4181	10733322
	3	17919	28	53757
	4	66	0	264
				10787343

DEVELOPER	2	5367	4201	10734146
	3	17519	42	52557
	4	56	0	224
	5	1	0	5
	10786932			
ABCDEFGH IJ	2	5368079	4146	10736158
	3	17297	49	51891
	4	62	1	248
	10788297			
EARLYWATCH	2	5363189	4181	10726378
	3	17405	28	52215
	4	50	1	200
	10778793			
12345678901	2	5371727	4160	10743454
	3	17377	46	52131
	4	62	0	248
	10795833			
ABCDEFGHIJK	2	5365705	4105	10731410
	3	17629	46	52887
	4	50	0	200
	10784497			
ABCDEFGHIJKL	2	5363917	4183	10727834
	3	17566	46	52698
	4	64	0	256
	10780788			
PASSWORDTEST	2	5364123	4164	10728246
	3	17603	33	52809
	4	51	0	204
	10781259			

Wesentlich gravierender sind die Auswirkungen, wenn das Passwort mit nicht zum 7bit-ASCII-Zeichensatz gehörenden Zeichen beginnt, bedingt durch die bei CODVN D verwendete UTF-8-Konvertierung.

Da im Hash-Algorithmus sporadisch nur die ersten (7 oder mehr) Bytes des Passwortes in die Hash-Berechnung eingehen, lassen sich etliche 8stellige Passworte finden, die mit 3 oder 4 nicht zum 7bit-ASCII-Zeichensatz gehörenden Zeichen beginnen und zu denen es 4stellige Passworte mit identischem Hashwert gibt.

(Nach UTF-8-Konvertierung müssen mindestens die ersten 7 Bytes des 4stelligen Passwortes mit den ersten 7 Bytes des nach UTF-8 konvertierten 8stelligen Passwortes übereinstimmen, damit eine Kollision möglich ist. Je mehr Bytes übereinstimmen, desto wahrscheinlicher werden Kollisionen.)

Für die auch in den vorhergehenden Beispielen aufgeführten Benutzernamen habe ich die Hashwerte zu allen $(10*10*10-10)*10 = 9900$ zulässigen 4stelligen Passwörtern ermittelt, die aus folgenden 10 Zeichen bestehen:

- ¡ (umgekehrtes !)
- § (Paragraph)
- ° (Grad)
- ² (hoch 2)
- ³ (hoch 3)
- μ (my)
- Ä (A-Umlaut)
- Ö (O-Umlaut)
- Ü (U-Umlaut)
- ß (Eszett)

Für die 4stelligen Passworte, zu denen es Hash-Kollisionen mit längeren Passwörtern gibt, habe ich die Anzahl von Hash-Kollisionen für 8stellige Passworte ermittelt, die in den ersten 4 Zeichen mit dem 4stelligen Passwort übereinstimmen und deren folgende 4 Zeichen nur aus den Ziffern 0-9 sowie den Buchstaben A-Z bestehen. (Pro Benutzer und 4stelligem Passwort wurden also $36*36*36*36=1.679.616$ 8stellige Passworte überprüft.)

Auch hier ergibt sich, wie schon bei der Prüfung auf Hash-Kollisionen von in den ersten 7 Zeichen übereinstimmenden Passwörtern, eine mit der Länge des Benutzernamens zunehmende Anzahl von Hash-Kollisionen, wobei ab einer Länge von mehr als 8 Zeichen keine weitere Zunahme zu erkennen ist.

In der folgenden Übersicht wurden die 4stelligen Passworte, für die nur die ersten 7 Bytes relevant sind, separat aufgeführt. (Die Spalte "Nur 7 Bytes relevant" ist mit X markiert.)

Für die entsprechenden Passworte treten Kollisionen auch mit Passwörtern auf, die nur in den ersten 3 Zeichen mit dem 4stelligen Passwort übereinstimmen, wenn nach UTF-8-Konvertierung des 4. Zeichens das erste Byte mit dem ersten Byte des nach UTF-8 konvertierten 4. Zeichens des 4stelligen Passworts übereinstimmt.

(Hinweis: das erste Byte eines nach UTF-8 konvertierten, nicht zum 7bit-ASCII-Zeichensatz gehörenden Zeichens ist für alle Unicode-Zeichen $\leq 0x00FF$ entweder $0xC2$ oder $0xC3$, s. <http://www.cl.cam.ac.uk/~mgk25/unicode.html#utf-8>)

Die Spalte "8stellige Passworte" enthält die Anzahl 8stelliger Passworte, deren Hashwert identisch mit dem Hashwert des auf 4 Zeichen gekürzten Passworts ist. Die Spalte "4stellige Passworte" enthält die Anzahl 4stelliger Passworte, deren Hashwert identisch mit dem Hashwert von mindestens einem 8stelligen, in den ersten 4 Zeichen übereinstimmenden Passwort ist.

Die Spalte "Maximale Anzahl Kollisionen" enthält die jeweils größte, zu einem 4stelligen Passwort gefundene Anzahl 8stelliger Passworte mit identischem Hashwert.

Die Spalte "Passwort" enthält das jeweils erste Passwort, für das die in der vorhergehenden Spalte angegebene Anzahl 8stelliger Passworte mit identischem Hashwert gefunden wurde.

Übersicht Hash-Kollisionen 8stelliger Passworte mit 4stelligen Passworten:

Benutzername	Nur 7 Bytes relevant	8stellige Passworte	4stellige Passworte	Maximale Anzahl Kollisionen	Passwort
123	-	1652	22	243	$3 \circ 2 i$
ABC	-	1208	22	224	$\S \ddot{O} i^2$
SAP	-	1205	14	236	$\mu \mu^2 \mu$
DDIC	X	10	1	10	$\ddot{O}^2 \circ 3$
DDIC	-	5518	63	267	$\ddot{A} \beta^3 \S$
SAP*	-	6558	56	433	$^2 \ddot{A}^3 \ddot{O}$
12345	X	267	5	92	$^0 \mu^2 \mu$
12345	-	25785	149	685	$\mu^0 2 \S$
ABCDE	X	64	3	33	$\mu \S^3 \beta$
ABCDE	-	23323	149	709	$i \ddot{O} \ddot{U}^3$
ABCDEF	X	1631	23	211	$\ddot{A} \ddot{U} \mu \ddot{U}$
ABCDEF	-	108906	381	928	$^3 \S \ddot{O}^3$
TMSADM	X	1359	18	232	$\S \ddot{O} \S \beta$
TMSADM	-	101637	397	894	$^3 i \mu i$
1234567	X	7786	57	483	$^2 \S^3 \beta$
1234567	-	292757	765	1018	$^3 3 \mu^0$
ABCDEFG	X	7306	58	427	$\ddot{U} \S^2 2$
ABCDEFG	-	314425	813	1013	$\mu \ddot{A}^2 i$
DITTRICH	X	7844	61	441	$\mu^0 2 \beta$
DITTRICH	-	676535	1424	1052	$i^3 2 i$
ABCDEFGH	X	6744	62	484	$\beta \ddot{A} \S^3$
ABCDEFGH	-	700242	1456	1078	$^3 3 \ddot{U}^2$
DEVELOPER	X	7848	70	443	$\S \S i \ddot{U}$
DEVELOPER	-	693425	1428	1067	$^2 \ddot{A} \ddot{O} \S$
ABCDEFGHIJ	X	8666	56	452	$\ddot{O}^2 \circ \ddot{U}$
ABCDEFGHIJ	-	695677	1420	1074	$\ddot{A} \ddot{O} \beta \ddot{A}$
EARLYWATCH	X	6387	59	446	$\ddot{O} \ddot{O}^2 \ddot{O}$
EARLYWATCH	-	680813	1413	1060	$\ddot{U}^2 \ddot{A} i$
12345678901	X	12056	74	475	$\ddot{O}^0 \circ \beta$
12345678901	-	684957	1413	1072	$^3 \mu \ddot{O} \mu$
ABCDEFGHIJK	X	8740	60	486	$\ddot{U} \ddot{U} \ddot{O} \mu$
ABCDEFGHIJK	-	706581	1479	1068	$^3 \circ i \beta$
ABCDEFGHIJKL	X	7864	63	440	$\beta i \mu \S$
ABCDEFGHIJKL	-	682540	1383	1068	$\S^3 \ddot{U}^0$
PASSWORDTEST	X	7704	58	460	$\ddot{U} \beta \S \ddot{O}$
PASSWORDTEST	-	683925	1395	1063	$^0 3 \ddot{U}^0$

Um auch die Verteilung der Wahrscheinlichkeit darzustellen, mit der Kollisionen auftreten, ist eine etwas detailliertere Übersicht nötig.

Die folgende Tabelle enthält pro Benutzer und statistisch wahrscheinlicher Anzahl von Passwörtern bis zum Finden einer Hash-Kollision die Anzahl 4stelliger Passwörter, für die entsprechende Kollisionen gefunden wurden.

Schon während der Berechnung des Hashwertes für das 4stellige Passwort steht fest, ob es zu Hash-Kollisionen mit längeren Passwörtern kommen kann und wie wahrscheinlich solche Kollisionen sind.

Daher ist eine Gruppierung entsprechend der Wahrscheinlichkeit von Kollisionen vorgenommen worden.

Bedeutung der einzelnen Spalten:

- Benutzername
SAP-Benutzername
- Kollision mit and. PW-Beginn möglich
Bei den mit X gekennzeichneten Zeilen gehen nur die ersten 7 Byte des nach UTF-8 konvertierten 4stelligen Passwortes in den Hashwert ein.
Daher treten mit der in der Tabelle angegebenen Wahrscheinlichkeit Kollisionen auch mit Passwörtern auf, die nur in den ersten 3 Zeichen mit dem 4stelligen Passwort übereinstimmen, wenn nach UTF-8-Konvertierung des 4. Zeichens das erste Byte mit dem ersten Byte des nach UTF-8 konvertierten 4. Zeichens des 4stelligen Passworts übereinstimmt. Für die anderen Zeilen treten nur bei Übereinstimmung der ersten 4 Zeichen Kollisionen auf.
(Das erste Byte nach UTF-8-Konvertierung ist entweder 0xC2 oder 0xC3.)
- Erwartete Anzahl PW für eine Kollision
Durchschnittliche Anzahl auszuprobierender 8stelliger Passwörter, damit eine Hash-Kollision mit dem 4stelligen Passwort gefunden wird
- Erwartete Anzahl Kollisionen
Nach der ermittelten statistischen Wahrscheinlichkeit zu erwartende Anzahl von Kollisionen mit dem Hashwert des 4stelligen Passworts bei insgesamt 1.679.616 überprüften 8stelligen Passwörtern
- Anzahl gefundener 4stelliger Passwörter
Anzahl Passwörter für den Benutzernamen, für die Kollisionen (mit der in der vorhergehenden Spalte angegebenen Wahrscheinlichkeit) auftreten
- Minimale Anzahl Kollisionen
Minimale tatsächlich ermittelte Anzahl Kollisionen für ein entsprechendes 4stelliges Passwort
- Maximale Anzahl Kollisionen
Maximale tatsächlich ermittelte Anzahl Kollisionen für ein entsprechendes 4stelliges Passwort

Benutzername	Kollisionen mit and. PW-Beginn möglich	Erwartete Anzahl PW für eine Kollision	Erwartete Anzahl Kollisionen	Anzahl gefundener 4stelliger Passworte	Minimale Anzahl Kollisionen	Maximale Anzahl Kollisionen
123	-	7490	224	2	206	243
123	-	17476	96	10	85	109
123	-	52429	32	6	25	34
123	-	262144	6	4	5	8
ABC	-	7490	224	1	224	224
ABC	-	17476	96	7	76	109
ABC	-	52429	32	12	20	33
ABC	-	262144	6	2	2	4
SAP	-	7490	224	3	203	236
SAP	-	17476	96	3	98	111
SAP	-	52429	32	7	21	37
SAP	-	262144	6	1	5	5
DDIC	-	7490	224	10	193	267
DDIC	-	17476	96	27	81	119
DDIC	-	52429	32	23	22	44
DDIC	X	262144	6	1	10	10
DDIC	-	262144	6	3	6	7
SAP*	-	4033	416	6	401	433
SAP*	-	7490	224	7	212	244
SAP*	-	17476	96	20	80	120
SAP*	-	52429	32	16	24	44
SAP*	-	262144	6	7	4	9
12345	-	2595	647	1	685	685
12345	-	4033	416	24	379	462
12345	-	7490	224	43	194	270
12345	X	17476	96	2	85	92
12345	-	17476	96	47	79	116
12345	X	26214	64	1	72	72
12345	-	52429	32	26	18	43
12345	X	131072	13	1	14	14
12345	X	262144	6	1	4	4
12345	-	262144	6	8	3	10
ABCDE	-	2595	647	3	633	709
ABCDE	-	4033	416	13	375	440
ABCDE	-	7490	224	48	195	263
ABCDE	-	17476	96	44	75	115
ABCDE	X	52429	32	2	26	33
ABCDE	-	52429	32	32	19	44
ABCDE	X	262144	6	1	5	5
ABCDE	-	262144	6	9	3	12
ABCDEF	-	1942	865	8	808	928
ABCDEF	-	2595	647	39	587	694
ABCDEF	-	4033	416	100	378	456
ABCDEF	-	7490	224	118	185	259
ABCDEF	X	8738	192	2	177	211
ABCDEF	X	17476	96	8	87	118
ABCDEF	-	17476	96	82	70	115
ABCDEF	X	26214	64	4	57	76
ABCDEF	X	52429	32	4	28	34
ABCDEF	-	52429	32	26	21	47
ABCDEF	X	131072	13	3	8	13

ABCDEF	X	262144	6	2	6	10
ABCDEF	-	262144	6	8	3	9
TMSADM	-	1942	865	3	842	894
TMSADM	-	2595	647	36	593	701
TMSADM	-	4033	416	92	342	461
TMSADM	X	7490	224	2	222	232
TMSADM	-	7490	224	119	184	258
TMSADM	X	17476	96	5	82	114
TMSADM	-	17476	96	102	70	120
TMSADM	X	26214	64	3	58	71
TMSADM	X	52429	32	6	30	39
TMSADM	-	52429	32	39	23	42
TMSADM	X	131072	13	1	14	14
TMSADM	X	262144	6	1	4	4
TMSADM	-	262144	6	6	2	7
1234567	-	1691	993	4	967	1018
1234567	-	1942	865	38	801	940
1234567	-	2595	647	159	587	713
1234567	X	3745	448	2	427	483
1234567	X	4033	416	2	411	426
1234567	-	4033	416	237	359	480
1234567	X	7490	224	13	199	245
1234567	-	7490	224	196	188	268
1234567	X	8738	192	6	177	231
1234567	X	17476	96	11	81	116
1234567	-	17476	96	99	71	117
1234567	X	26214	64	11	51	74
1234567	X	52429	32	7	20	40
1234567	-	52429	32	32	20	47
1234567	X	131072	13	4	10	17
1234567	X	262144	6	1	6	6
ABCDEF	-	1691	993	7	967	1013
ABCDEF	-	1942	865	53	799	915
ABCDEF	-	2595	647	150	588	709
ABCDEF	X	3745	448	1	427	427
ABCDEF	X	4033	416	1	419	419
ABCDEF	-	4033	416	253	369	483
ABCDEF	X	7490	224	10	212	246
ABCDEF	-	7490	224	207	191	264
ABCDEF	X	8738	192	13	170	224
ABCDEF	X	17476	96	8	82	106
ABCDEF	-	17476	96	112	72	125
ABCDEF	X	26214	64	6	46	72
ABCDEF	X	52429	32	11	26	47
ABCDEF	-	52429	32	29	23	42
ABCDEF	X	131072	13	8	9	14
ABCDEF	-	262144	6	2	8	9
DITTRICH	-	1691	993	48	926	1052
DITTRICH	-	1942	865	189	787	946
DITTRICH	-	2595	647	339	565	701
DITTRICH	X	3745	448	2	438	441
DITTRICH	X	4033	416	2	422	428
DITTRICH	-	4033	416	407	360	471
DITTRICH	X	7490	224	10	207	244
DITTRICH	-	7490	224	287	170	266
DITTRICH	X	8738	192	7	181	211
DITTRICH	X	17476	96	15	77	115
DITTRICH	-	17476	96	118	76	117
DITTRICH	X	26214	64	13	56	77

DITTRICH	X	52429	32	4	21	34
DITTRICH	-	52429	32	34	16	53
DITTRICH	X	131072	13	6	6	21
DITTRICH	X	262144	6	2	4	6
DITTRICH	-	262144	6	2	6	6
ABCDEFGHI	-	1691	993	65	932	1078
ABCDEFGHI	-	1942	865	191	783	948
ABCDEFGHI	-	2595	647	346	579	714
ABCDEFGHI	X	3745	448	1	484	484
ABCDEFGHI	X	4033	416	2	434	471
ABCDEFGHI	-	4033	416	399	363	475
ABCDEFGHI	X	7490	224	5	189	263
ABCDEFGHI	-	7490	224	300	189	277
ABCDEFGHI	X	8738	192	10	170	209
ABCDEFGHI	X	17476	96	11	85	114
ABCDEFGHI	-	17476	96	125	75	125
ABCDEFGHI	X	26214	64	15	49	86
ABCDEFGHI	X	52429	32	7	22	40
ABCDEFGHI	-	52429	32	26	19	44
ABCDEFGHI	X	131072	13	9	10	20
ABCDEFGHI	X	262144	6	2	3	10
ABCDEFGHI	-	262144	6	4	8	16
DEVELOPER	-	1691	993	50	928	1067
DEVELOPER	-	1942	865	180	767	934
DEVELOPER	-	2595	647	389	565	717
DEVELOPER	X	3745	448	2	424	443
DEVELOPER	X	4033	416	1	443	443
DEVELOPER	-	4033	416	405	358	489
DEVELOPER	X	7490	224	6	205	249
DEVELOPER	-	7490	224	245	174	268
DEVELOPER	X	8738	192	11	165	215
DEVELOPER	X	17476	96	15	82	109
DEVELOPER	-	17476	96	125	73	130
DEVELOPER	X	26214	64	21	54	80
DEVELOPER	X	52429	32	7	19	38
DEVELOPER	-	52429	32	33	20	40
DEVELOPER	X	131072	13	3	13	21
DEVELOPER	X	262144	6	4	4	11
DEVELOPER	-	262144	6	1	7	7
ABCDEFGHIJ	-	1691	993	67	933	1074
ABCDEFGHIJ	-	1942	865	197	788	954
ABCDEFGHIJ	-	2595	647	344	567	715
ABCDEFGHIJ	X	4033	416	6	416	452
ABCDEFGHIJ	-	4033	416	391	357	476
ABCDEFGHIJ	X	7490	224	12	196	264
ABCDEFGHIJ	-	7490	224	269	186	260
ABCDEFGHIJ	X	8738	192	8	180	216
ABCDEFGHIJ	X	17476	96	8	74	120
ABCDEFGHIJ	-	17476	96	121	73	125
ABCDEFGHIJ	X	26214	64	15	48	71
ABCDEFGHIJ	X	52429	32	3	29	39
ABCDEFGHIJ	-	52429	32	25	23	45
ABCDEFGHIJ	X	131072	13	4	9	16
ABCDEFGHIJ	-	262144	6	6	3	11
EARLYWATCH	-	1691	993	49	924	1060
EARLYWATCH	-	1942	865	181	787	934
EARLYWATCH	-	2595	647	385	577	720
EARLYWATCH	X	4033	416	3	392	446
EARLYWATCH	-	4033	416	367	357	471

EARLYWATCH	X	7490	224	7	209	239
EARLYWATCH	-	7490	224	271	182	262
EARLYWATCH	X	8738	192	8	168	214
EARLYWATCH	X	17476	96	9	82	119
EARLYWATCH	-	17476	96	121	75	122
EARLYWATCH	X	26214	64	10	48	82
EARLYWATCH	X	52429	32	11	24	45
EARLYWATCH	-	52429	32	38	19	39
EARLYWATCH	X	131072	13	10	9	19
EARLYWATCH	X	262144	6	1	5	5
EARLYWATCH	-	262144	6	1	2	2
12345678901	-	1691	993	62	937	1072
12345678901	-	1942	865	179	785	954
12345678901	-	2595	647	343	579	743
12345678901	X	3745	448	7	420	475
12345678901	X	4033	416	3	371	455
12345678901	-	4033	416	423	358	478
12345678901	X	7490	224	12	206	253
12345678901	-	7490	224	265	187	272
12345678901	X	8738	192	12	168	218
12345678901	X	17476	96	13	86	119
12345678901	-	17476	96	110	74	121
12345678901	X	26214	64	15	50	81
12345678901	X	52429	32	8	22	44
12345678901	-	52429	32	27	22	44
12345678901	X	131072	13	4	8	20
12345678901	-	262144	6	4	3	9
ABCDEFGHIJK	-	1691	993	48	926	1068
ABCDEFGHIJK	-	1942	865	192	786	940
ABCDEFGHIJK	-	2595	647	378	584	715
ABCDEFGHIJK	X	3745	448	3	482	486
ABCDEFGHIJK	X	4033	416	4	369	422
ABCDEFGHIJK	-	4033	416	401	362	495
ABCDEFGHIJK	X	7490	224	9	201	242
ABCDEFGHIJK	-	7490	224	293	183	265
ABCDEFGHIJK	X	8738	192	8	174	205
ABCDEFGHIJK	X	17476	96	12	87	114
ABCDEFGHIJK	-	17476	96	143	68	122
ABCDEFGHIJK	X	26214	64	11	55	85
ABCDEFGHIJK	X	52429	32	8	26	37
ABCDEFGHIJK	-	52429	32	21	23	46
ABCDEFGHIJK	X	131072	13	3	10	11
ABCDEFGHIJK	X	262144	6	2	5	5
ABCDEFGHIJK	-	262144	6	3	5	9
ABCDEFGHIJKL	-	1691	993	55	942	1068
ABCDEFGHIJKL	-	1942	865	210	783	957
ABCDEFGHIJKL	-	2595	647	333	578	706
ABCDEFGHIJKL	X	4033	416	4	381	440
ABCDEFGHIJKL	-	4033	416	389	363	478
ABCDEFGHIJKL	X	7490	224	11	192	249
ABCDEFGHIJKL	-	7490	224	255	181	269
ABCDEFGHIJKL	X	8738	192	6	171	200
ABCDEFGHIJKL	X	17476	96	18	81	127
ABCDEFGHIJKL	-	17476	96	107	73	133
ABCDEFGHIJKL	X	26214	64	9	59	74
ABCDEFGHIJKL	X	52429	32	3	26	32
ABCDEFGHIJKL	-	52429	32	33	23	41
ABCDEFGHIJKL	X	131072	13	9	14	22
ABCDEFGHIJKL	X	262144	6	3	4	10

ABCDEFGHIJKL	-	262144	6	1	9	9
PASSWORDTEST	-	1691	993	64	931	1063
PASSWORDTEST	-	1942	865	182	789	938
PASSWORDTEST	-	2595	647	342	583	745
PASSWORDTEST	X	3745	448	1	460	460
PASSWORDTEST	X	4033	416	2	406	406
PASSWORDTEST	-	4033	416	417	370	474
PASSWORDTEST	X	7490	224	11	202	244
PASSWORDTEST	-	7490	224	265	187	265
PASSWORDTEST	X	8738	192	10	162	215
PASSWORDTEST	X	17476	96	13	72	123
PASSWORDTEST	-	17476	96	96	72	119
PASSWORDTEST	X	26214	64	10	51	75
PASSWORDTEST	X	52429	32	5	25	43
PASSWORDTEST	-	52429	32	26	18	46
PASSWORDTEST	X	131072	13	6	10	17
PASSWORDTEST	-	262144	6	3	6	12

In einem weiteren Test wurde ermittelt, wie wahrscheinlich Hash-Kollisionen sind, wenn ein Passwort mit entsprechend vielen Zeichen beginnt, die nach UTF-8-Konvertierung durch 2 Bytes repräsentiert werden.

Wegen der für diesen Test wesentlich längeren Rechenzeit war ein Test für alle in den vorhergehenden Tests verwendeten Benutzernamen nicht möglich.

Es wurden für alle mit der Zeichenfolge ÄÖÜß beginnenden Passworte die Hashwerte für den Benutzer EARLYWATCH ermittelt, einmal nur unter Verwendung von 7bit-ASCII-Zeichen, einmal indem zusätzlich 66 nicht zum 7bit-ASCII-Zeichensatz gehörende Zeichen berücksichtigt wurden.

Bei Beschränkung auf 7bit-ASCII-Zeichen wurden also für $69 \cdot 69 \cdot 69 \cdot 69 = 22.676.121$ Passworte die Hashwerte ermittelt und gezählt, wie oft jeder einzelne Hashwert vorkam.

Es wurden insgesamt nur 3.948.620 verschiedene Hashwerte ermittelt.

Nur für 2.982.019 Passworte (also etwa 13% aller getesteten Passworte) kam es nicht zu Hash-Kollisionen.

Dies waren die 10 am häufigsten vorkommenden Hashwerte (jeweils mit der Anzahl Passworte):

- 13.716 CC7C3927947F6848
- 13.574 63866900B8A6C403
- 13.528 16AEC08FA2E8CCA2
- 13.501 7918A48DD7A31928
- 13.481 DA0C035FFC8FD61F
- 13.313 B4171234E8F64C76
- 13.235 663749E904F38773
- 13.174 26C8E9350E4AD104
- 13.161 A1102636BD7C27F4
- 13.133 3E1B1D2FBB6A8AD0

Bei Berücksichtigung nicht zum 7bit-ASCII-Zeichensatz gehörender Zeichen wurden insgesamt $135 \cdot 135 \cdot 135 \cdot 135 = 332.150.625$ Passworte überprüft.

Es wurden 13.207.570 verschiedene Hashwerte ermittelt.

Nur bei 6.891.312 Passworten (also nur etwa 2% aller Passworte) kam es zu keiner Kollision.

Die 10 am häufigsten vorkommenden Hashwerte waren:

- 197.087 B4171234E8F64C76
- 197.005 63866900B8A6C403
- 196.604 DA0C035FFC8FD61F
- 196.445 7918A48DD7A31928
- 196.404 CC7C3927947F6848
- 196.320 16AEC08FA2E8CCA2
- 196.233 3E1B1D2FBB6A8AD0
- 195.970 663749E904F38773
- 195.966 26C8E9350E4AD104
- 195.243 A1102636BD7C27F4

Um zu prüfen, wie häufig Kollisionen vorkommen, wenn nur drei der ersten 4 Zeichen nicht zum 7bit-ASCII-Zeichensatz gehören, wurden für den Benutzer EARLYWATCH alle 332.150.625 Passworte überprüft, die mit ÄÖÜ- beginnen.

Hier wurden 50.060.494 verschiedene Hashwerte ermittelt.

Bei 28.066.400 Passworten (ca. 8,4% der Passworte) kam es zu keiner Kollision.

Die 10 am häufigsten vorkommenden Hashwerte waren:

- 88.787 6ECA62BAA9883964
- 83.390 8AA14A27662A21A7
- 49.645 97B466B1B84E2E58
- 49.563 1DF375E132B970D5
- 49.533 F0CD906185B6C117
- 49.475 D3620A483BE80467
- 49.436 FF14E984436A6FAB
- 49.360 2F9D7A22188D3D9D
- 49.311 01AB6B9C7D8E734E
- 49.277 C715A197EC05C78F

Die für den neu eingeführten Algorithmus (CODVN D) auftretenden Kollisionen bei Verwendung von nicht zum 7bit-ASCII-Zeichensatz gehörenden Zeichen sprechen gegen eine Änderung des Profile-Parameters login/password_charset auf 2 und damit die Nutzung von CODVN D zur Ermittlung der Passwort-Hashwerte.

Ausblick

Es stellt sich die Frage, warum der Nachfolger von CODVN B nicht C, sondern D heißt. Als CODVN C ist eventuell der 2002 von Wolfgang Janzen in <http://groups.google.com/groups?selm=3D384CAE.7E251CC1%40sap.com> erwähnte Hash-Algorithmus unter Verwendung von SHA-1 statt MD5 in Arbeit. Auch im OSS-Hinweis 2467 wird erwähnt, dass zukünftig SHA-1 als Hash-Algorithmus verwendet werden soll.

Zwei Jahre sollten eigentlich für die Implementierung eines neuen und sicheren Passwort-Algorithmus ausreichen.

Ein längerer Hashwert trägt zur Verringerung der Anzahl von Hash-Kollisionen allerdings nichts bei, wenn bei der Implementierung des Hash-Algorithmus ähnlich gravierende Fehler wie in den bisherigen Hash-Algorithmen gemacht werden.

Außerdem sollte unbedingt dafür gesorgt werden, dass die zur Ermittlung eines Hashwertes benötigte Zeit wesentlich größer als bisher ist, damit Dictionary- bzw. Brute-Force-Attacken erschwert werden.

Da SAP-Anwender inzwischen für nahezu alle anderen Anwendungen bei der Eingabe von Passwörtern die Groß- bzw. Kleinschreibung beachten müssen, sollte dies auch bei der Implementierung des neuen Hash-Algorithmus für SAP-Passwörter in Erwägung gezogen werden.

Frank Dittrich