CRIPTORETO 2. LA FRUTA DE VERANO NUNCA MADURA EN OTOÑO
(Continuación del Criptoreto 1)

Con pista de 26x62 valores

Publicación del reto: 6 de septiembre de 2013 a las 12:00 horas

Publicación de primeras pistas: 25 de septiembre de 2013 a las 12:00 horas

RESUELTO POR DANI TORREGROSA, DE BILBAO, ESPAÑA, TRAS 20 DÍAS Y 8 HORAS

¿Crees que la parte que tiene más tropa gana? Entonces es sólo una cuestión de acudir a la batalla basándose en la cuenta de los soldados. ¿Crees que la parte con más riqueza gana? Entonces es sólo una cuestión de acudir a la batalla basándose en la medida del grano. ¿Piensas que la parte con armas más afiladas o armaduras más fuertes gana? Entonces sería fácil determinar al vencedor. En consecuencia, los ricos no están forzosamente seguros, los pobres no están necesariamente inseguros, la mayoría no prevalece necesariamente, las minorías no fracasan forzosamente. Lo que determina quién gana y quién pierde, quién está seguro y quién en peligro, es su ciencia…

Sun Bin. El arte de la guerra

El verano pasó incesantemente con la satisfacción de los analistas del GCHQ, liderados por el capitán Mr. Beaumont, de haber descifrado una comunicación del mismo terminal del presidente ruso Vladimir Putin. El gobierno británico pudo analizar con tiempo, junto con sus socios norteamericanos, los peligros de lo que se confirmaría más tarde como el asilo de Edward Snowden por parte del gobierno ruso. No obstante, no todo fueron alegrías, parte de la red de inteligencia británica desplegada en Rusia fue descubierta y expulsada del país por espionaje. Parece ser que el agente doble que consiguió la información en Rusia levantó las sospechas de los grupos de contrainteligencia. Varios miembros de una red de espionaje más amplia fueron investigados y expulsados del país.

En este clima de hostilidad el GCHQ fue presionado para recopilar la suficiente información para acelerar una operación en curso de espionaje en territorio británico. Un supuesto agente ruso intercambiaba información cifrada con otros supuestos agentes mediante un terminal particular. Se sabe que ese terminal utilizaba un algoritmo RSA y claves de 8192 bits con validez de la clave de una semana. Cada clave es generada por el agente. La última clave pública que se dispone para ese terminal es de 8192 bits:

e=17
n=00955e655158204f2970de1eef8f1d6d302a9e84cef94219c77f2e23ab326b460eeb380c3c2f
65301779de8c2df0f705ac5e55daf378f7dd7c7b2e92f58e9b0f23756bbb93cb8ba20d7e043eb71
52689ea68c2dd7feb11a9b16b38a96ba18e237306e9ca3f698ef58688b0b08fadae203c48627a3
74b2deb25f6efaf94d2c65e715d6b0054557d65e315ffead60fe49924848173d346a6e1eaaf28cae
4985ed2cda304365dccff1e794978c5e091ad89488717e6b2f2d689088ce138c80db93ca6ed968
51c3e6e2180390e9c4bbe48d578d113be2a683b0045c4272fc67e92c9d3cc7563584fa649dddc8
ceddf00ec10dbd57496b94a2efddf623690d4b103acd43d09e97a2ebfc8915c0677eddf18a9927a
e240b6d044dbe9233a699271018100891196624 64c299a8c88bd81e951a6bf693bb2cacfd1486
d67f8458c45529a864bc2d1709d81f9262c1a381109506cdd25cb9b01a1aef6586378e8c4915b3f

26985790fbcbb82331999cc1aa6716b415a8c81cb57f69465f6c3f4d39368e40e7f9d2425ba62de
207e0135bb640266ded9b3f07dc98d04c5dc91fbe8c6d44336c3256eb182fa46c6f67b4e244ee5a
f1474dc324d27b8d8fc5b9fbbf7be801ce40add18b43d9444bd1dd0f548cacc402337fa4989e7f82
deaf2c6aa8656d1a5ff1d90ea713ba037a7bd96abae8b95066d9798090e75fdd94434ad88ae562
3053426b76391f79531976f30fb7583061b4fa2b97b3b39c553dbee426eac35d509e48fca718418
da78d329d674d5a7d4e7353aa03e1f2749dc52808194d7cd95ccd984d7c16e81ca141f2659c127
8dbda5d42c09d06c29784377fb62d377287dd8e5b1b2f4ef0f199b5abf38ec53100ffe66bf5524bef
acd8ebdd6add57dc13d25d82647c5aab28be8680ae3b891bd094ed62a18bc2a028f46ab94616a
712d533472af3efd77aed3204096cc7ac8a8a49808f60d599e0385c578c59d1565b310bb76e7c8
220cc9de3cd1fc1f71d6a6163059c0fb44927b21bd77dcf1dd9e289143824b20e5ae057d4afd1ad
84f5804d2f17c7d6f05a1d1d59735ca0ddfc11f13a4a737bb84ec8307b9c0b1a9215ec701ea7913
7881ab18f2b5b02c360003113b74841e765cc791f3d8618edc4ce4e981e63c477d942b7d70ce2d
fe01d2ad1336d88705d0fa86d62873eb2c376e9072ea81b82ec7229e33ddc2dbc0073046b065b
b770dcdb865904bb6a3e89a05619da8bb562514122f152187d7cbf8328c64c90cdd28205d28f4b
2c029972c28dd2508dde3e2562c888e4593eeed9d1f3ac959d6634556bb5690d94e2fb50a6bce
bc1d590f7673e0722bbb0a7179c907191f184c5cff56d14fe4775acc6f47949cc0b02b98bc731aad
3f22477be6f434372aafa7402ff2027d461fa9dd4cb3d98da998bb63e3ea5ada67b5119e58bad1b
27153a749

Además se conoce que tiene un receptor de radiofrecuencia por el que, hasta lo que se conoce, lo utiliza para recibir una serie de números sin sentido aparente. No se sabe si esta información tiene algún tipo de valor o si se recibe más información por otras frecuencias no detectadas o mediante canales ocultos.

De momento solo se ha podido interceptar los siguientes 26x62 valores (véase ANEXO)

Da la sensación que el supuesto agente va a dejar su apartamento alquilado. ¿Quizás sospeche que está siendo monitorizado? La última comunicación cifrada interceptada entre el agente bajo vigilancia y el receptor/es de la comunicación es la siguiente:

C=13040651349013124501769431990889129847809022923431467748414330391929572038
32202067239942507308175019995540464064874323725193864031239106725918864162 83
82386685582542540836604906988135875197891783265469365259542353836242253612 60
89972600087464258136301502822515391960777512948324325932262304698516842308 91
12405153159517232671381217382180877318558083364208140600828983012821297219 83
18684793812689952142306839190090179243142503016288717886281337650422297208 45
01156068662646393071532686584400538553861188502225136473854227980589211921 15
96190824971142907056543532709250473544995758725260183990562479632037578239 28
65513936322274406594110438952103580341886789380774667682044036186743132830 30
97410668556251641416830666745000208810069279975372589024272630635594483274 63
19218931039801446290294853747660950000589687575864499412026382937245424966 60
96023194942984228117250332261587260843

¿Qué información estará comunicando el agente? ¿Qué recomendaría hacer? El tiempo corre en su contra…

Desing by @mindcrypt

ANEXO. 26x62 valores

1000|1063|1126|1189|1252|1315|1378|1441|1504|1567|1630|1693|1756|1819|1882|1945|2008|2071|2134|2197|2260|2323|2386|2449|2512|2575|2638|2701|2764|2827|2890|2953|3016|3079|3142|3205|3268|3331|3394|3457|3520|3583|3646|3709|3772|3835|3898|3961|4024|4087|4150|4213|4276|4339|4402|4465|4528|4591|4654|4717|4780|4843|

1001|1065|1129|1193|1257|1321|1385|1449|1513|1577|1641|1705|1769|1833|1897|1961|2025|2089|2153|2217|2281|2345|2409|2473|2537|2601|2665|2729|2793|2857|2921|2923|2987|3051|3115|3179|3243|3307|3371|3435|3499|3563|3627|3691|3755|3819|3883|3947|4011|4075|4139|4203|4267|4331|4395|4459|4523|4587|4651|4715|4779|4782|

1002|1067|1132|1197|1262|1327|1392|1457|1522|1587|1652|1717|1782|1847|1912|1977|2042|2107|2172|2237|2240|2305|2370|2435|2500|2565|2630|2695|2760|2825|2891|2956|3021|3086|3151|3216|3281|3346|3411|3476|3541|3544|3609|3674|3739|3804|3869|3934|3999|4064|4129|4194|4259|4324|4389|4454|4519|4584|4649|4714|4781|4783|

1003|1069|1135|1201|1267|1333|1399|1465|1531|1597|1663|1729|1795|1861|1927|1931|1997|2063|2129|2195|2262|2328|2394|2460|2526|2592|2658|2724|2790|2856|2861|2927|2993|3059|3125|3191|3257|3323|3389|3455|3522|3588|3654|3720|3786|3790|3856|3922|3988|4054|4120|4186|4252|4318|4384|4450|4516|4582|4648|4716|4720|4784|

1004|1071|1138|1205|1272|1339|1406|1473|1540|1607|1674|1741|1746|1813|1880|1948|2015|2082|2149|2216|2284|2351|2418|2485|2490|2557|2624|2691|2758|2826|2894|2961|3028|3095|3162|3229|3234|3301|3368|3436|3503|3570|3637|3704|3771|3839|3906|3973|3978|4045|4112|4179|4246|4313|4380|4447|4514|4581|4650|4718|4721|4785|

1005|1073|1141|1209|1277|1345|1413|1481|1549|1617|1623|1691|1760|1828|1896|1965|2033|2101|2169|2238|2244|2312|2380|2448|2517|2585|2653|2721|2789|2859|2865|2933|3001|3069|3137|3206|3274|3342|3410|3479|3485|3553|3621|3689|3758|3826|3894|3963|4031|4099|4105|4173|4241|4309|4377|4445|4513|4583|4652|4719|4722|4786|

1006|1075|1144|1213|1282|1351|1420|1489|1496|1565|1635|1704|1774|1843|1913|1982|2051|2058|2127|2198|2267|2336|2405|2475|2544|2551|2620|2689|2759|2830|2899|2968|3037|3106|3113|3183|3252|3321|3391|3461|3530|3599|3606|3676|3745|3814|3884|3953|4022|4092|4161|4168|4237|4306|4375|4444|4515|4585|4653|4658|4723|4787|

1007|1077|1147|1217|1287|1357|1427|1435|1506|1576|1647|1718|1788|1858|1929|1937|2007|2078|2148|2220|2290|2360|2368|2439|2509|2580|2650|2720|2792|2801|2871|2941|3011|3082|3153|3223|3293|3302|3373|3443|3513|3584|3655|3725|3733|3803|3874|3944|4015|4085|4156|4164|4234|4304|4374|4446|4517|4586|4655|4659|4724|4788|

1008|1079|1150|1221|1292|1363|1372|1444|1516|1588|1659|1731|1802|1811|1884|1955|2027|2098|2170|2180|2251|2322|2395|2466|2538|2609|2618|2690|2763|2835|2906|2977|2986|3058|3130|3201|3273|3345|3417|3426|3497|3569|3641|3713|3784|3794|3865|3937|4008|4080|4152|4223|4232|4303|4376|4448|4518|4588|4656|4660|4725|4789|

1009|1081|1153|1225|1297|1369|1380|1453|1526|1599|1671|1682|1754|1827|1901|1973|2046|2056|2130|2203|2275|2348|2421|2431|2504|2577|2649|2723|2797|2807|2879|2951|3025|3098|3108|3181|3254|3327|3400|3472|3483|3556|3629|3701|3775|3847|3858|3930|4003|4076|4148|4221|4231|4305|4378|4449|4520|4589|4657|4661|4726|4790|

1010|1083|1156|1229|1302|1313|1388|1462|1536|1610|1621|1696|1770|1844|1918|1991|2003|2077|2152|2226|2299|2311|2385|2459|2533|2607|2619|2694|2769|2842|2915|2928|3002|3075|3149|3224|3236|3310|3383|3458|3532|3543|3617|3692|3765|3840|3913|3925|3998|4072|4146|4220|4233|4307|4379|4451|4521|4590|4596|4662|4727|4791|

1011|1085|1159|1233|1307|1320|1396|1471|1546|1559|1634|1710|1785|1860|1873|1949|2023|2099|2175|2187|2263|2338|2413|2426|2501|2576|2652|2728|2741|2815|2889|2966|3041|3054|3129|3204|3280|3355|3367|3444|3518|3594|3607|3682|3757|3832|3908|3920|3995|4070|4145|4222|4235|4308|4381|4452|4522|4592|4597|4663|4728|4792|

1012|1087|1162|1237|1250|1328|1404|1480|1556|1571|1648|1724|1800|1815|1891|1968|2044|2059|2136|2211|2288|2302|2378|2455|2531|2608|2623|2700|2776|2851|2867|2943|3019|3096|3110|3187|3263|3339|3416|3430|3507|3582|3660|3673|3750|3827|3903|3917|3993|4069|4147|4162|4236|4310|4382|4453|4524|4593|4598|4664|4729|4793|

1013|1089|1165|1241|1256|1335|1412|1490|1505|1583|1661|1738|1753|1832|1909|1987|2002|2081|2159|2235|2252|2330|2407|2484|2499|2579|2657|2735|2749|2828|2905|2982|2998|3076|3155|3170|3247|3325|3403|3419|3495|3574|3651|3667|3744|3822|3900|3915|3992|4071|4149|4163|4238|4311|4383|4455|4525|4594|4599|4665|4730|4794|

1014|1091|1168|1245|1263|1342|1421|1437|1517|1595|1675|1690|1771|1850|1928|1944|2024|2104|2120|2200|2278|2357|2374|2453|2532|2550|2629|2708|2785|2804|2882|2962|3040|3057|3136|3215|3232|3312|3390|3469|3487|3565|3644|3723|3740|3818|3897|3914|3994|4073|4151|4165|4239|4312|4385|4456|4526|4595|4600|4666|4731|4795|

1015|1093|1171|1187|1269|1349|1429|1447|1528|1608|1626|1707|1787|1867|1886|1966|2047|2065|2144|2225|2243|2324|2403|2483|2503|2584|2664|2681|2762|2843|2860|2940|3022|3102|3120|3199|3282|3298|3379|3460|3539|3558|3638|3718|3736|3816|3896|3916|3996|4074|4153|4166|4240|4314|4386|4457|4527|4534|4601|4667|4732|4796|

1016|1095|1174|1192|1275|1356|1375|1458|1539|1558|1640|1722|1804|1823|1905|1986|2006|2088|2168|2189|2271|2353|2372|2454|2536|2556|2637|2717|2739|2819|2902|2983|3004|3085|3166|3186|3267|3350|3369|3450|3533|3551|3633|3715|3734|3815|3899|3918|3997|4077|4154|4167|4242|4315|4387|4458|4529|4535|4602|4668|4733|4797|

1017|1097|1177|1198|1281|1364|1384|1468|1551|1572|1655|1737|1759|1841|1924|1946|2030|2112|2133|2215|2298|2319|2402|2487|2508|2591|2673|2696|2778|2799|2881|2965|2985|3068|3152|3173|3256|3338|3360|3442|3526|3547|3630|3712|3732|3817|3901|3919|4000|4078|4155|4169|4243|4316|4388|4460|4530|4536|4603|4669|4734|4798|

1018|1099|1180|1203|1288|1371|1394|1478|1500|1585|1669|1692|1777|1862|1883|1969|2053|2074|2160|2182|2266|2
350|2373|2458|2543|2564|2647|2734|2754|2840|2863|2947|3032|3053|3139|3222|3245|3330|3414|3437|3521|3542|36
26|3711|3735|3820|3902|3921|4001|4079|4157|4170|4244|4317|4390|4461|4531|4537|4604|4670|4735|4799|

1019|1101|1183|1208|1294|1317|1403|1488|1512|1600|1622|1709|1794|1818|1904|1990|2014|2100|2123|2209|2295|2
318|2406|2430|2516|2600|2626|2711|2736|2820|2908|2931|3015|3103|3126|3212|3237|3320|3407|3431|3516|3602|36
25|3714|3737|3821|3904|3923|4002|4081|4158|4171|4245|4319|4391|4462|4532|4538|4605|4671|4736|4800|

1020|1103|1124|1214|1300|1325|1414|1438|1525|1613|1638|1726|1750|1838|1925|1952|2038|2064|2151|2239|2264|2
352|2377|2465|2489|2574|2666|2687|2777|2803|2888|2976|3003|3090|3116|3202|3289|3315|3402|3427|3514|3601|36
28|3716|3738|3823|3905|3924|4004|4082|4159|4172|4247|4320|4392|4463|4533|4539|4606|4672|4737|4801|

1021|1105|1128|1219|1306|1334|1423|1450|1538|1564|1654|1743|1768|1857|1887|1975|2000|2091|2117|2206|2294|2
321|2412|2438|2525|2554|2642|2732|2756|2847|2874|2963|2990|3078|3167|3194|3284|3309|3398|3424|3512|3603|36
31|3717|3741|3824|3907|3926|4005|4083|4160|4174|4248|4321|4393|4464|4472|4540|4607|4673|4738|4802|

1022|1107|1133|1224|1251|1343|1432|1461|1552|1580|1670|1699|1790|1817|1908|1936|2028|2054|2145|2234|2265|2
356|2384|2474|2502|2594|2621|2712|2742|2832|2919|2949|3039|3067|3159|3188|3277|3305|3396|3423|3515|3545|36
32|3719|3742|3825|3909|3927|4006|4084|4100|4175|4249|4322|4394|4466|4473|4541|4608|4674|4739|4803|

1023|1109|1137|1230|1259|1352|1381|1474|1502|1594|1625|1716|1747|1839|1870|1960|1992|2084|2176|2205|2297|2
329|2420|2447|2542|2570|2663|2693|2784|2814|2909|2937|3030|3061|3150|3180|3272|3303|3395|3425|3517|3546|36
34|3721|3743|3828|3910|3928|4007|4086|4101|4176|4250|4323|4396|4467|4474|4542|4609|4675|4740|4804|

1024|1111|1142|1235|1266|1360|1391|1485|1518|1611|1643|1735|1766|1863|1893|1985|2018|2111|2142|2236|2270|2
363|2393|2427|2520|2552|2644|2677|2771|2802|2896|2926|3020|3050|3145|3176|3270|3300|3397|3428|3519|3548|36
35|3722|3746|3829|3911|3929|4009|4088|4102|4177|4251|4325|4397|4468|4475|4543|4610|4676|4741|4805|

1025|1113|1146|1240|1274|1368|1402|1434|1532|1563|1660|1694|1789|1822|1917|1951|2045|2079|2174|2208|2242|2
337|2367|2467|2496|2595|2628|2722|2753|2850|2884|2979|3012|3046|3141|3174|3269|3304|3399|3429|3523|3549|36
36|3724|3747|3830|3912|3931|4010|4089|4103|4178|4253|4326|4398|4469|4476|4544|4611|4677|4742|4806|

Reconocimientos:

1. El ganador del reto será aquel que obtenga el mensaje en claro y documente brevemente el procedimiento seguido.

2. El ganador del reto obtendrá una copia gratuita del libro "Cifrado de las comunicaciones digitales. De la cifra clásica al algoritmo RSA" publicado por la editorial 0xWORD, así como será considerada para futuros proyectos con la red Criptored.

http://0xword.com/libros/36-libro-cifrado-comunicaciones-rsa.html

3. Dudas/comentarios: cryptoreto@gmail.com / Twitter @mindcrypt

4. Síguenos en Twitter: @criptored @mindcrypt

5. El Criptoreto1 puede ser consultado en la siguiente dirección:
http://www.criptored.upm.es/paginas/criptoretoRSAjulio2013.pdf

Resultados y Comentarios del Segundo Criptoreto

El presente texto recoge las soluciones al criptoreto de septiembre de 2013, lanzado en Internet por la Red Temática Criptored el martes 2 de julio de 2013.

Dani Torregrosa, de Bilbao, España, ha sido la primera persona en resolver el segundo criptoreto de la red temática Criptored, tardando 20 días y 8 horas.

A continuación se adjunta la explicación breve del diseño del reto y seguidamente se adjunta la solución de Dani Torregrosa.

[Explicación @mindcrypt]

El presente reto presenta un escenario real en el cual una información a proteger se codifica previamente con un mecanismo desconocido y finalmente se cifra con un algoritmo criptográfico estándar. En tanto en cuanto el mecanismo desconocido se mantenga en secreto puede proteger temporalmente frente a roturas o malos usos del algoritmo criptográfico estándar.

En este problema tiene dos fases:

- La utilización del algoritmo RSA de manera insegura. Si m^e < N, entonces es posible realizar la raíz n-ésima y recuperar m. Al hacer esto con el Criptoreto se observa un texto sin sentido que da pistas para considerar el resto de números emitidos por la emisora.

- La codificación está basada en un cifrador por homófonos de 2 orden. Se forma una matriz de 62x62 valores (letras minúsculas, mayúsculas y números). Cada valor de la matriz está formado por un número de 4 dígitos. Dado un número en la matriz recuperando el índice y la columna se obtienen por cada número 2 caracteres del mensaje original.

- La generación de los números, y por tanto de la matriz, sigue la siguiente lógica:

Tenemos 62 filas de 62 valores cada una.

1000 a 1061
1062 a 1123
1124 a 1185
….

Generamos una nueva matriz mediante nuevas filas que consisten en ir rotando a la izquierda para cada valor de cada fila la posición de la fila actual. Con ese resultado se construye la matriz final conmutando filas por columnas.

Por ejemplo,

Fila 0: roto 0 posiciones a la izquierda: 1000 a 1061
Fila 1: roto 1 posiciones a la izquierda: 1063,1065, 1067,…, 1090

La matriz final se construye conmutando filas por columna.

Como se puede observar en este ejemplo utilizando criptografía por homófonos aunque los valores sean “predecibles” la cosa se complica bastante.

Agradecer nuevamente la excelente acogida a este reto, con más de 1.500 descargas, cordiales saludos y hasta el próximo criptoreto.

Madrid, 9 de octubre de 2013

Dr. Alfonso Muñoz
Dr. Jorge Ramió
Editores de Criptored

[Solución de Dani Torregrosa]

De los enunciados de los retos (este es la segunda parte de otro anterior), se tiene que:

El dispositivo de comunicación usa 2 cifrados, uno a continuación de otro: uno es RSA y el otro es desconocido.

En el cifrado RSA se usa:

- El módulo N de la clave RSA es de 8192 bits.

- e = 17

- El texto cifrado C es:
13040651349013124501769431990889129847809022923431467748414330391929572038322020672399425073081750199955404640648743237251938640312391067259188641628382386685582542540836604906988135875197891783265469365259542353836242253612608997260008746425813630150282251539196077751294832432593226230469851684230891124051531595172326713812173821808773185580833642081406008289830128212972198318684793812689952142306839190090179243142503016288717886281337650422297208450115606866264639307153268658440053855386118850222513647385422798058921192115961908249711429070565435327092504735449957587252601839905624796320375782392865513936322274406594110438952103580341886789380774667682044036186743132830309741066855625164141683066674500020881006927997537258902427263063559448327463192189310398014462902948537476609500005896875758644994120263829372454249666096023194942984228117250332261587260843

- Hay unos listado de números capturados a través de una emisora de radio, que tienen relación con la segunda fase del cifrado.

Se supone el orden siguiente al aplicar los 2 cifrados:

mensaje (m) => (cifrado desconocido) => c1 => (RSA) => C

Fase primera del descifrado: cifrado RSA

Mirando los datos (módulo, exponente y texto cifrado), y sus tamaños relativos, parece posible que:

C = c1^e < N

en cuyo caso se puede descifrar fácilmente sin necesidad de tener la clave privada, sin más que extrayendo la "raíz e" de C.

Se realiza una prueba, usando python y python-gmpy, y viendo el resultado:

```
$ python
>>> import gmpy
>>>                                  c                                  =
13040651349013124501769431990889129847809022923431467748414330391929572038322020672399425073081750199955404640648743237251938640312391067259188641628382386685582542540836604906988135875197891783265469365259542353836242253612608997260008746425813630150282251539196077751294832432593226230469851684230891124051531595172326713812173821808773185580833642081406008289830128212972198318684793812689952142306839190090179243142503016288717886281337650422297208450115606866264639307153268658440053855386118850222513647385422798058921192115961908249711429070565435327092504735449957587252601839905624796320375782392865513936322274406594110438952103580341886789380774667682044036186743132830309741066855625164141683066674500020881006927997537258902427263063559448327463192189310398014462902948537476609500005896875758644994120263829372454249666096023194942984228117250332261587260843
```

```
203832202067239942507308175019995540464064874323725193864031239106725918864162838238668558254254083660490698813587519789178326546936525954235383624225361260899726000874642581363015028225153919607775129483243259322623046985168423089112405153159517232671381217382180877318558083364208140600828983012821297219831868479381268995214230683919009017924314250301628871788628133765042229720845011560686626463930715326865844005385538611885022251364738542279805892119211596190824971142907056543532709250473544995758725260183990562479632037578239286551393632227440659411043895210358034188678938077466768204403618674313283030974106685562516414168306667450002088100692799753725890242726306355944832746319218931039801446290294853747660950000589687575864499412026382937245424966609602319494298422811725033226158726084 3L
>>> gmpy.root(c,17)
(mpz(17461294150514244339179010191281195511321003153616 43L), 1)
```

se confirma (el 1 del final indica esto) que C era una potencia 17 perfecta, algo poco probable para números tan grandes, y que nos indica que vamos por el buen camino con bastante probabilidad.

Con este paso:

- Se refuerza la validez de la hipótesis del orden de los cifrados:
  mensaje (m) => (cifrado desconocido) => c1 => (RSA) => C

- Se obtiene c1 = 17461294150514244339179010191281195511321003153616 43

NOTA: para evitar este fallo (y otros) del mecanismo RSA "puro", el espía debería haber usado padding al usar RSA.
(https://en.wikipedia.org/wiki/RSA_%28algorithm%29#Padding_schemes)

Fase segunda del descifrado: cifrado desconocido

Esta fase es más complicada, más de prueba-error, al ser el cifrado desconocido, y al ser el texto cifrado bastante corto.

Se supone que los datos de los números recibidos en la radio tienen relación con este cifrado.

En un primer momento sólo se disponía de una fila de números recibida.

La fila era una serie aritmética:

```
[1000, 1063, 1126, 1189, 1252, 1315, 1378, 1441, 1504, 1567, 1630,
1693, 1756, 1819, 1882, 1945, 2008, 2071, 2134, 2197, 2260, 2323,
2386, 2449, 2512, 2575, 2638, 2701, 2764, 2827, 2890, 2953, 3016,
3079, 3142, 3205, 3268, 3331, 3394, 3457, 3520, 3583, 3646, 3709,
3772, 3835, 3898, 3961, 4024, 4087, 4150, 4213, 4276, 4339, 4402,
4465, 4528, 4591, 4654, 4717, 4780, 4843]
```

Equivalente a la serie:

```
[1000 + 63*x for x in xrange(62)]
```

Se trata de números de 4 cifras.

Dividiendo c1 en grupos de 4 cifras,

```
c1=17461294150514244339179010191281195511321003153616 43
c1s=str(c1)
c1=map(int,[c1s[x:x+4] for x in xrange(0,len(c1s),4)]); c1
[1746, 1294, 1505, 1424, 4339, 1790, 1019, 1281, 1955, 1132, 1003,
1536, 1643]
```

Se ven indicios que refuerzan que vamos por buen camino:

- Todos los números empiezan por 1, salvo uno (el 4339).

- El número que no empieza por 1 (el 4339), está en el listado de los números recibidos en la emisora.

A partir de aquí, siguen una serie de pruebas y errores, que no llevan a ningún lado, hasta que publican la primera pista.

Con la primera pista, amplían los números recibidos en la emisora.

Ahora son 26x62 números.

El número 26 puede ser el número de caracteres en un abecedario: `abcdefghijklmnopqrstuvwxyz`

El número 62 puede ser el número de caracteres en un abecedario extendido con minúsculas, mayúsculas, números (podría ser este orden u otro):

```
abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789
```

Con esto se puede pensar en los números recibidos como en una matriz, en cuyas columnas como cabecera se ponen los 62 caracteres (a-zA-Z0-9), y en las filas los 26 caracteres a-z.

Buscando los números de c1 (ya separados de 4 en 4) en la matriz, vemos que se encuentran todos en la matriz dada excepto uno (el 1424).

a) Primera hipótesis:

El mensaje se extrae buscando los números en la matriz, y viendo la letra cabecera de fila.

el numero 1424 no se ha encontrado en la matriz

```
>>>m
'etn*awtricdky'
```

No parece tener mucho sentido.
[El código para extraerlo es el de la tercera hipótesis un poco modificado.]

b) Segunda hipótesis:

El mensaje se extrae buscando los números en la matriz, y viendo la letra cabecera de columna.

el numero 1424 no se ha encontrado en la matriz
```
>>m
'mei*1maepcaik'
```

Tampoco tiene mucho sentido
[El código para extraerlo es el de la tercera hipótesis un poco modificado.]

c) Tercera hipótesis:
La idea feliz fue juntar los dos primeros intentos: Cada número encontrado en la matriz, representa 2 letras, indicadas por la fila y la columna.

```
$ python
>>> abc26 = 'abcdefghijklmnopqrstuvwxyz'
>>>                                abc62                                  =
'abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789
'
>>> c1 = [1746, 1294, 1505, 1424, 4339, 1790, 1019, 1281, 1955, 1132,
1003,
 1536, 1643]
>>> m = ''
>>> for numero in c1:
...   encontrado = 0
...   for linea in matriz:
...     for num in linea:
...       if numero == num:
...         encontrado = encontrado + 1
...         columna = linea.index(num)
...         fila = matriz.index(linea)
...         m = m + abc62[columna] + abc26[fila]
...   if encontrado == 0:
...     print "el numero",numero,"no se ha encontrado en la matriz"
...     m = m + "**"
...       elif  encontrado  >  1:  print  "el  numero",numero,"se  ha
encontrado",enco
ntrado,"veces"
...
el numero 1424 no se ha encontrado en la matriz
print m
meetin**1amwaterpiccadikky
```

Esto ya empieza a tener sentido.

Faltan 2 detalles por corregir:

- Parece que hay una pequeña sustitución en el abecedario, entre la 'l' y la 'k'

- Nos faltan las letras representadas por el número 1424

Para encontrar el 1424 en la matriz, que no está en la parte de la matriz capturada, parece que hay que extender la matriz.

Tiene sentido que la matriz real sea de 62x62.

Además parece que está generada con una fórmula y se puede recalcular.

Estudiando la matriz, se ve que las columnas están formadas por 62 números consecutivos.

La primera del 1000 al 1061, la segunda del 1062 al 1123, ... columna = (numero-1000)//62 [primera columna es la 0]

Así es fácil ver que el 1424 está en la columna de la letra 'g', que se adapta bien a lo que sabíamos del mensaje, formando 'meeting'

meeting*1amwaterpiccadikky

Falta la fila. Se ven patrones, aunque no he conseguido la fórmula para ubicar un número en una fila, pero el único carácter del abecedario elegido que tiene sentido en el mensaje sería un 1:

Con esto, quedaría como mensaje: "meeting11amwaterpicadilly"

Con esta información, recomendaría al capitán Mr. Beaumont estar atentos a las 11 a.m., en todos los lugares que relacionen "water" y "piccadilly", p.ej., la fuente de la plaza Piccadilly Circus.

2013-09-26
Dani Torregrosa, danitorwS
w0pr

Anexo: Matriz de números recibidos en emisora

matriz = [[1000, 1063, 1126, 1189, 1252, 1315, 1378, 1441, 1504, 1567,
1630, 1693, 1756, 1819, 1882, 1945, 2008, 2071, 2134, 2197, 2260, 2323, 238
6, 2449, 2512, 2575, 2638, 2701, 2764, 2827, 2890, 2953, 3016, 3079, 3142,
3205, 3268, 3331, 3394, 3457, 3520, 3583, 3646, 3709, 3772, 3835, 3898, 396
1, 4024, 4087, 4150, 4213, 4276, 4339, 4402, 4465, 4528, 4591, 4654, 4717,
4780, 4843], [1001, 1065, 1129, 1193, 1257, 1321, 1385, 1449, 1513, 1577, 1
641, 1705, 1769, 1833, 1897, 1961, 2025, 2089, 2153, 2217, 2281, 2345, 2409
, 2473, 2537, 2601, 2665, 2729, 2793, 2857, 2921, 2923, 2987, 3051, 3115, 3
179, 3243, 3307, 3371, 3435, 3499, 3563, 3627, 3691, 3755, 3819, 3883, 3947
, 4011, 4075, 4139, 4203, 4267, 4331, 4395, 4459, 4523, 4587, 4651, 4715, 4
779, 4782], [1002, 1067, 1132, 1197, 1262, 1327, 1392, 1457, 1522, 1587, 16
52, 1717, 1782, 1847, 1912, 1977, 2042, 2107, 2172, 2237, 2240, 2305, 2370,
2435, 2500, 2565, 2630, 2695, 2760, 2825, 2891, 2956, 3021, 3086, 3151, 32
16, 3281, 3346, 3411, 3476, 3541, 3544, 3609, 3674, 3739, 3804, 3869, 3934,
3999, 4064, 4129, 4194, 4259, 4324, 4389, 4454, 4519, 4584, 4649, 4714, 47
81, 4783], [1003, 1069, 1135, 1201, 1267, 1333, 1399, 1465, 1531, 1597, 166
3, 1729, 1795, 1861, 1927, 1931, 1997, 2063, 2129, 2195, 2262, 2328, 2394,
2460, 2526, 2592, 2658, 2724, 2790, 2856, 2861, 2927, 2993, 3059, 3125, 319
1, 3257, 3323, 3389, 3455, 3522, 3588, 3654, 3720, 3786, 3790, 3856, 3922,
3988, 4054, 4120, 4186, 4252, 4318, 4384, 4450, 4516, 4582, 4648, 4716, 472
0, 4784], [1004, 1071, 1138, 1205, 1272, 1339, 1406, 1473, 1540, 1607, 1674

, 1741, 1746, 1813, 1880, 1948, 2015, 2082, 2149, 2216, 2284, 2351, 2418, 2
485, 2490, 2557, 2624, 2691, 2758, 2826, 2894, 2961, 3028, 3095, 3162, 3229
, 3234, 3301, 3368, 3436, 3503, 3570, 3637, 3704, 3771, 3839, 3906, 3973, 3
978, 4045, 4112, 4179, 4246, 4313, 4380, 4447, 4514, 4581, 4650, 4718, 4721
, 4785], [1005, 1073, 1141, 1209, 1277, 1345, 1413, 1481, 1549, 1617, 1623,
1691, 1760, 1828, 1896, 1965, 2033, 2101, 2169, 2238, 2244, 2312, 2380, 24
48, 2517, 2585, 2653, 2721, 2789, 2859, 2865, 2933, 3001, 3069, 3137, 3206,
3274, 3342, 3410, 3479, 3485, 3553, 3621, 3689, 3758, 3826, 3894, 3963, 40
31, 4099, 4105, 4173, 4241, 4309, 4377, 4445, 4513, 4583, 4652, 4719, 4722,
4786], [1006, 1075, 1144, 1213, 1282, 1351, 1420, 1489, 1496, 1565, 1635,
1704, 1774, 1843, 1913, 1982, 2051, 2058, 2127, 2198, 2267, 2336, 2405, 247
5, 2544, 2551, 2620, 2689, 2759, 2830, 2899, 2968, 3037, 3106, 3113, 3183,
3252, 3321, 3391, 3461, 3530, 3599, 3606, 3676, 3745, 3814, 3884, 3953, 402
2, 4092, 4161, 4168, 4237, 4306, 4375, 4444, 4515, 4585, 4653, 4658, 4723,
4787], [1007, 1077, 1147, 1217, 1287, 1357, 1427, 1435, 1506, 1576, 1647, 1
718, 1788, 1858, 1929, 1937, 2007, 2078, 2148, 2220, 2290, 2360, 2368, 2439
, 2509, 2580, 2650, 2720, 2792, 2801, 2871, 2941, 3011, 3082, 3153, 3223, 3
293, 3302, 3373, 3443, 3513, 3584, 3655, 3725, 3733, 3803, 3874, 3944, 4015
, 4085, 4156, 4164, 4234, 4304, 4374, 4446, 4517, 4586, 4655, 4659, 4724, 4
788], [1008, 1079, 1150, 1221, 1292, 1363, 1372, 1444, 1516, 1588, 1659, 17
31, 1802, 1811, 1884, 1955, 2027, 2098, 2170, 2180, 2251, 2322, 2395, 2466,
2538, 2609, 2618, 2690, 2763, 2835, 2906, 2977, 2986, 3058, 3130, 3201, 32
73, 3345, 3417, 3426, 3497, 3569, 3641, 3713, 3784, 3794, 3865, 3937, 4008,
4080, 4152, 4223, 4232, 4303, 4376, 4448, 4518, 4588, 4656, 4660, 4725, 47
89], [1009, 1081, 1153, 1225, 1297, 1369, 1380, 1453, 1526, 1599, 1671, 168
2, 1754, 1827, 1901, 1973, 2046, 2056, 2130, 2203, 2275, 2348, 2421, 2431,
2504, 2577, 2649, 2723, 2797, 2807, 2879, 2951, 3025, 3098, 3108, 3181, 325
4, 3327, 3400, 3472, 3483, 3556, 3629, 3701, 3775, 3847, 3858, 3930, 4003,
4076, 4148, 4221, 4231, 4305, 4378, 4449, 4520, 4589, 4657, 4661, 4726, 479
0], [1010, 1083, 1156, 1229, 1302, 1313, 1388, 1462, 1536, 1610, 1621, 1696
, 1770, 1844, 1918, 1991, 2003, 2077, 2152, 2226, 2299, 2311, 2385, 2459, 2
533, 2607, 2619, 2694, 2769, 2842, 2915, 2928, 3002, 3075, 3149, 3224, 3236
, 3310, 3383, 3458, 3532, 3543, 3617, 3692, 3765, 3840, 3913, 3925, 3998, 4
072, 4146, 4220, 4233, 4307, 4379, 4451, 4521, 4590, 4596, 4662, 4727, 4791
], [1011, 1085, 1159, 1233, 1307, 1320, 1396, 1471, 1546, 1559, 1634, 1710,
1785, 1860, 1873, 1949, 2023, 2099, 2175, 2187, 2263, 2338, 2413, 2426, 25
01, 2576, 2652, 2728, 2741, 2815, 2889, 2966, 3041, 3054, 3129, 3204, 3280,
3355, 3367, 3444, 3518, 3594, 3607, 3682, 3757, 3832, 3908, 3920, 3995, 40
70, 4145, 4222, 4235, 4308, 4381, 4452, 4522, 4592, 4597, 4663, 4728, 4792]
, [1012, 1087, 1162, 1237, 1250, 1328, 1404, 1480, 1556, 1571, 1648, 1724,
1800, 1815, 1891, 1968, 2044, 2059, 2136, 2211, 2288, 2302, 2378, 2455, 253
1, 2608, 2623, 2700, 2776, 2851, 2867, 2943, 3019, 3096, 3110, 3187, 3263,
3339, 3416, 3430, 3507, 3582, 3660, 3673, 3750, 3827, 3903, 3917, 3993, 406
9, 4147, 4162, 4236, 4310, 4382, 4453, 4524, 4593, 4598, 4664, 4729, 4793],
[1013, 1089, 1165, 1241, 1256, 1335, 1412, 1490, 1505, 1583, 1661, 1738, 1
753, 1832, 1909, 1987, 2002, 2081, 2159, 2235, 2252, 2330, 2407, 2484, 2499
, 2579, 2657, 2735, 2749, 2828, 2905, 2982, 2998, 3076, 3155, 3170, 3247, 3
325, 3403, 3419, 3495, 3574, 3651, 3667, 3744, 3822, 3900, 3915, 3992, 4071
, 4149, 4163, 4238, 4311, 4383, 4455, 4525, 4594, 4599, 4665, 4730, 4794],
[1014, 1091, 1168, 1245, 1263, 1342, 1421, 1437, 1517, 1595, 1675, 1690, 17

71, 1850, 1928, 1944, 2024, 2104, 2120, 2200, 2278, 2357, 2374, 2453, 2532, 2550, 2629, 2708, 2785, 2804, 2882, 2962, 3040, 3057, 3136, 3215, 3232, 3312, 3390, 3469, 3487, 3565, 3644, 3723, 3740, 3818, 3897, 3914, 3994, 4073, 4151, 4165, 4239, 4312, 4385, 4456, 4526, 4595, 4600, 4666, 4731, 4795], [1015, 1093, 1171, 1187, 1269, 1349, 1429, 1447, 1528, 1608, 1626, 1707, 1787, 1867, 1886, 1966, 2047, 2065, 2144, 2225, 2243, 2324, 2403, 2483, 2503, 2584, 2664, 2681, 2762, 2843, 2860, 2940, 3022, 3102, 3120, 3199, 3282, 3298, 3379, 3460, 3539, 3558, 3638, 3718, 3736, 3816, 3896, 3916, 3996, 4074, 4153, 4166, 4240, 4314, 4386, 4457, 4527, 4534, 4601, 4667, 4732, 4796], [1016, 1095, 1174, 1192, 1275, 1356, 1375, 1458, 1539, 1558, 1640, 1722, 1804, 1823, 1905, 1986, 2006, 2088, 2168, 2189, 2271, 2353, 2372, 2454, 2536, 2556, 2637, 2717, 2739, 2819, 2902, 2983, 3004, 3085, 3166, 3186, 3267, 3350, 3369, 3450, 3533, 3551, 3633, 3715, 3734, 3815, 3899, 3918, 3997, 4077, 4154, 4167, 4242, 4315, 4387, 4458, 4529, 4535, 4602, 4668, 4733, 4797], [1017, 1097, 1177, 1198, 1281, 1364, 1384, 1468, 1551, 1572, 1655, 1737, 1759, 1841, 1924, 1946, 2030, 2112, 2133, 2215, 2298, 2319, 2402, 2487, 2508, 2591, 2673, 2696, 2778, 2799, 2881, 2965, 2985, 3068, 3152, 3173, 3256, 3338, 3360, 3442, 3526, 3547, 3630, 3712, 3732, 3817, 3901, 3919, 4000, 4078, 4155, 4169, 4243, 4316, 4388, 4460, 4530, 4536, 4603, 4669, 4734, 4798], [1018, 1099, 1180, 1203, 1288, 1371, 1394, 1478, 1500, 1585, 1669, 1692, 1777, 1862, 1883, 1969, 2053, 2074, 2160, 2182, 2266, 2350, 2373, 2458, 2543, 2564, 2647, 2734, 2754, 2840, 2863, 2947, 3032, 3053, 3139, 3222, 3245, 3330, 3414, 3437, 3521, 3542, 3626, 3711, 3735, 3820, 3902, 3921, 4001, 4079, 4157, 4170, 4244, 4317, 4390, 4461, 4531, 4537, 4604, 4670, 4735, 4799], [1019, 1101, 1183, 1208, 1294, 1317, 1403, 1488, 1512, 1600, 1622, 1709, 1794, 1818, 1904, 1990, 2014, 2100, 2123, 2209, 2295, 2318, 2406, 2430, 2516, 2600, 2626, 2711, 2736, 2820, 2908, 2931, 3015, 3103, 3126, 3212, 3237, 3320, 3407, 3431, 3516, 3602, 3625, 3714, 3737, 3821, 3904, 3923, 4002, 4081, 4158, 4171, 4245, 4319, 4391, 4462, 4532, 4538, 4605, 4671, 4736, 4800], [1020, 1103, 1124, 1214, 1300, 1325, 1414, 1438, 1525, 1613, 1638, 1726, 1750, 1838, 1925, 1952, 2038, 2064, 2151, 2239, 2264, 2352, 2377, 2465, 2489, 2574, 2666, 2687, 2777, 2803, 2888, 2976, 3003, 3090, 3116, 3202, 3289, 3315, 3402, 3427, 3514, 3601, 3628, 3716, 3738, 3823, 3905, 3924, 4004, 4082, 4159, 4172, 4247, 4320, 4392, 4463, 4533, 4539, 4606, 4672, 4737, 4801], [1021, 1105, 1128, 1219, 1306, 1334, 1423, 1450, 1538, 1564, 1654, 1743, 1768, 1857, 1887, 1975, 2000, 2091, 2117, 2206, 2294, 2321, 2412, 2438, 2525, 2554, 2642, 2732, 2756, 2847, 2874, 2963, 2990, 3078, 3167, 3194, 3284, 3309, 3398, 3424, 3512, 3603, 3631, 3717, 3741, 3824, 3907, 3926, 4005, 4083, 4160, 4174, 4248, 4321, 4393, 4464, 4472, 4540, 4607, 4673, 4738, 4802], [1022, 1107, 1133, 1224, 1251, 1343, 1432, 1461, 1552, 1580, 1670, 1699, 1790, 1817, 1908, 1936, 2028, 2054, 2145, 2234, 2265, 2356, 2384, 2474, 2502, 2594, 2621, 2712, 2742, 2832, 2919, 2949, 3039, 3067, 3159, 3188, 3277, 3305, 3396, 3423, 3515, 3545, 3632, 3719, 3742, 3825, 3909, 3927, 4006, 4084, 4100, 4175, 4249, 4322, 4394, 4466, 4473, 4541, 4608, 4674, 4739, 4803], [1023, 1109, 1137, 1230, 1259, 1352, 1381, 1474, 1502, 1594, 1625, 1716, 1747, 1839, 1870, 1960, 1992, 2084, 2176, 2205, 2297, 2329, 2420, 2447, 2542, 2570, 2663, 2693, 2784, 2814, 2909, 2937, 3030, 3061, 3150, 3180, 3272, 3303, 3395, 3425, 3517, 3546, 3634, 3721, 3743, 3828, 3910, 3928, 4007, 4086, 4101, 4176, 4250, 4323, 4396, 4467, 4474, 4542, 4609, 4675, 4740, 4804], [1024, 1111, 1142, 1235, 1266, 1360, 1391, 1485, 1518, 1611, 1643, 1735, 1766, 1863,

1893, 1985, 2018, 2111, 2142, 2236, 2270, 2363, 2393, 2427, 2520, 2552, 264
4, 2677, 2771, 2802, 2896, 2926, 3020, 3050, 3145, 3176, 3270, 3300, 3397,
3428, 3519, 3548, 3635, 3722, 3746, 3829, 3911, 3929, 4009, 4088, 4102, 417
7, 4251, 4325, 4397, 4468, 4475, 4543, 4610, 4676, 4741, 4805], [1025, 1113
, 1146, 1240, 1274, 1368, 1402, 1434, 1532, 1563, 1660, 1694, 1789, 1822, 1
917, 1951, 2045, 2079, 2174, 2208, 2242, 2337, 2367, 2467, 2496, 2595, 2628
, 2722, 2753, 2850, 2884, 2979, 3012, 3046, 3141, 3174, 3269, 3304, 3399, 3
429, 3523, 3549, 3636, 3724, 3747, 3830, 3912, 3931, 4010, 4089, 4103, 4178
, 4253, 4326, 4398, 4469, 4476, 4544, 4611, 4677, 4742, 4806]]